USCEE Report 341

UNIVERSITY OF SOUTHERN CALIFORNIA

TRANSFORM PROCESSING AND CODING OF IMAGES

Final Report

by

William K. Pratt
Harry C. Andrews

for

California Institute of Technology
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103

Contract No. 952312

ELECTRONIC SCIENCES LABORATORY

USC Engineering

March 1969

USCEE Report 341

TRANSFORM PROCESSING AND CODING OF IMAGES

Final Report

by

William K. Pratt
Harry C. Andrews

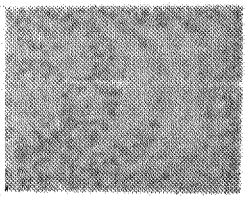
for

California Institute of Technology
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103

Contract No. 952312

Electronic Sciences Laboratory
University of Southern California
University Park
Los Angeles, California 90007

This work was performed for the Jet Propulsion Laboratory, California Institute of Technology, sponsored by the National Aeronautics and Space Administration under Contract NAS7-100.



TRANSFORM PROCESSING AND CODING OF IMAGES

Contents

Preface

This report comprises the final report for a study entitled, "Fourier Transform Processing and Coding of Images," performed by the University of Southern California, Electronic Sciences Laboratory for the Jet Propulsion Laboratory under the project direction of Fred C. Billingsley and Thomas Rindfleisch of JPL.

1. Introduction

The classic problem in the design of image coding systems for digital communication links is the search for a coding method which will minimize the number of code symbols required to describe an image. This coding method must not degrade the quality of the image beyond certain fidelity limits, and furthermore, the coding method must not be overly sensitive to channel errors. A great amount of investigation has been performed in the search for such image coding systems [1-3]. Unfortunately, most of the systems developed either do not exhibit satisfactory performance, or are too difficult to implement. The transform image coding method discussed in this report is a new approach to the problem of image coding. This image coding system achieves a reasonably large bandwidth reduction and offers a certain immunity to channel errors without significant image degradation. Implementation is presently feasible for slow scan image systems.

1.1 Background

The introduction of the fast Fourier transform algorithm [4-8] has led to the investigation of the Fourier transform image coding technique whereby the two dimensional Fourier transform of an image is transmitted over a channel rather than the image itself [9-12]. This investigation has itself led to the study of a related image coding technique in which an image is transformed by a Hadamard matrix operator [13-15]. The Hadamard matrix is a square array of plus and minus ones whose

rows and columns are orthogonal to one another. A high speed Hadamard transform computational algorithm, similar to the fast Fourier transform algorithm, has been developed [13].

The Fourier and Hadamard transforms are but two of a large number of transforms that have potential applications for image coding. In this report the general properties of image transforms are considered. Specific experimental examples of the performance of the Fourier and Hadamard image transforms are given.

1.2 Image Transform Coding

Figure 1-1 illustrates the block diagram of a generalized transform image coding system. In this system a transform is performed on the intensity samples of the image. The image transform samples are then quantized and coded for transmission over a digital link. At the receiver the received data is decoded, and an inverse transform is performed to reconstruct the original image. In principle the transforms could be implemented by optical, electrical, or digital techniques. The experimental results presented in this report have been obtained for a general purpose digital computer implementation of the image transforms. No attempt has been made to determine the "best" means of transform implementation, other than to present the most efficient computer algorithms.

As a prelude to the subsequent sections Figures 1-2 to 1-7 contain examples of the two dimensional Fourier and Hadamard transforms of an image. The original images containing 256 by 256 elements and linearly

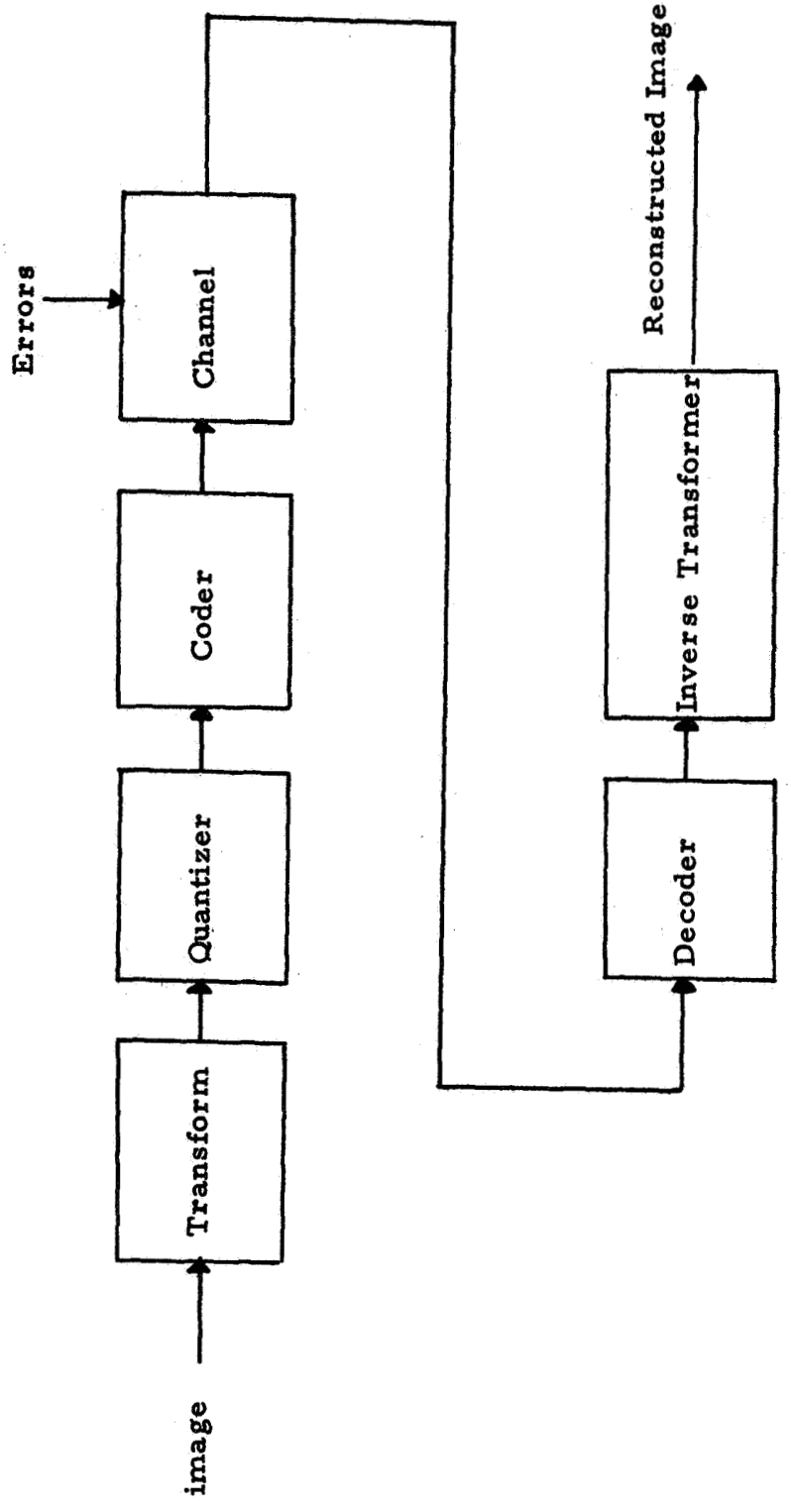
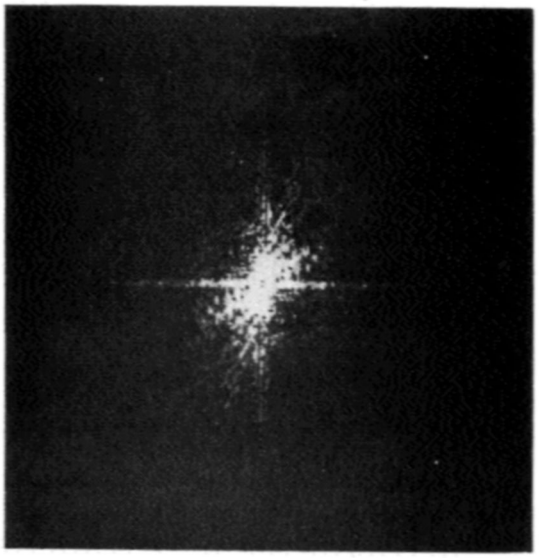


Figure 1-1. Image Transform Coding System

a) Original

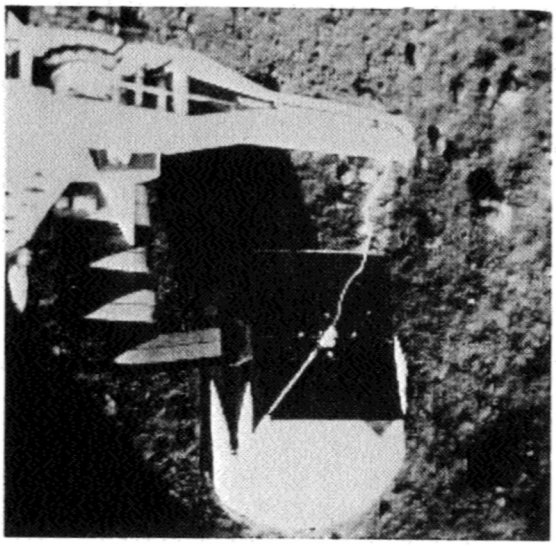
b) Inverse Fourier Transform of Fourier Transform

c) Magnitude of Fourier Transform

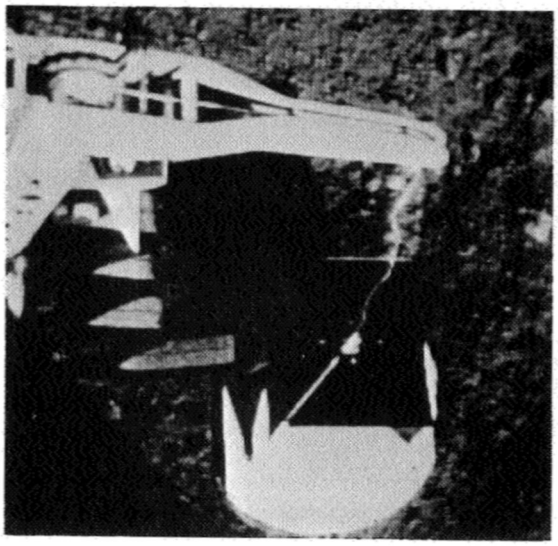


d) Logarithm of Magnitude of Fourier Transform

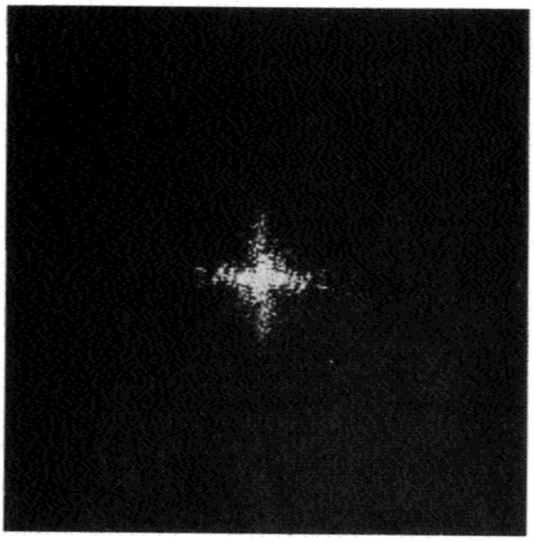
Figure 1-2. Fourier Transforms of Surveyor Spacecraft Footpad



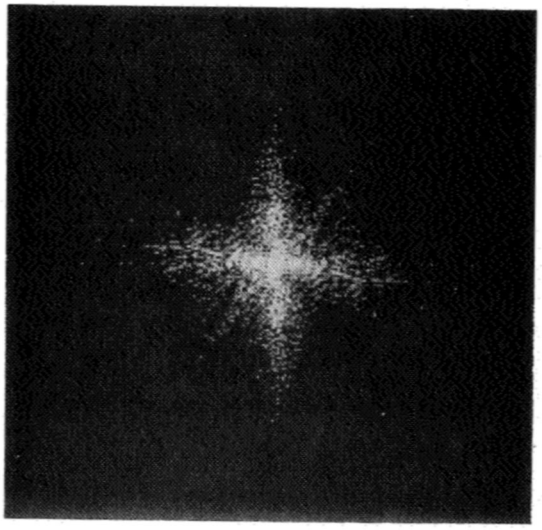
a) Original



b) Inverse Fourier Transform of Fourier Transform

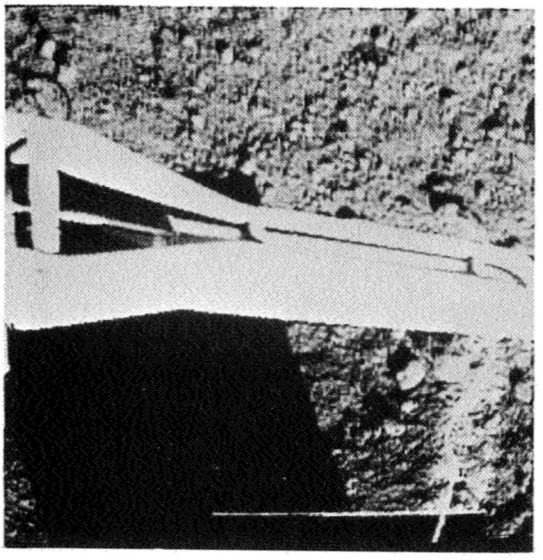


c) Magnitude of Fourier Transform

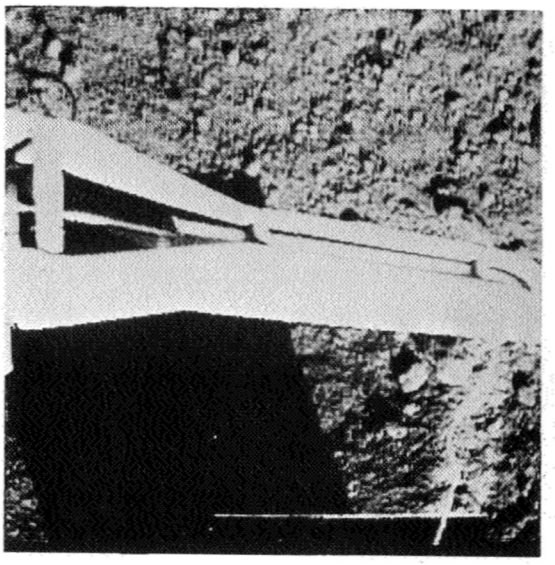


d) Logarithm of Magnitude of Fourier Transform

Figure 1-3. Fourier Transforms of Surveyor Spacecraft Experimental Box

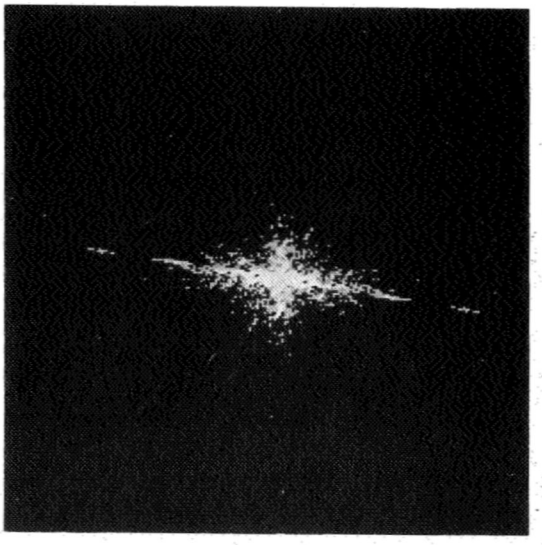


a) Original



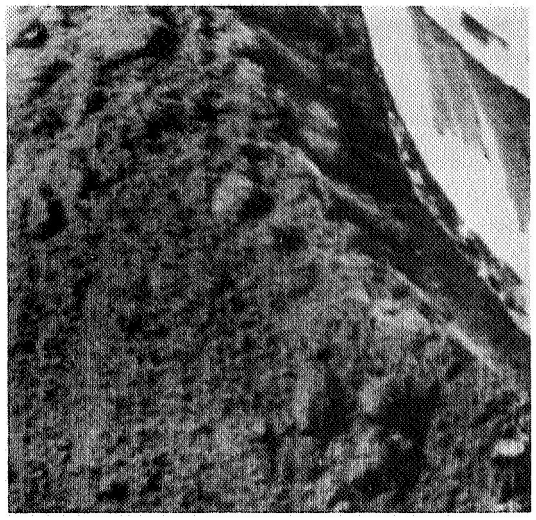
b) Inverse Fourier Transform of Fourier Transform

c) Magnitude of Fourier Transform

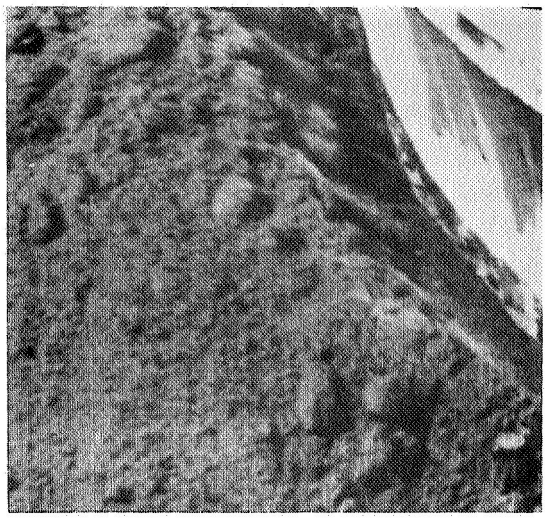


d) Logarithm of Magnitude of Fourier Transform

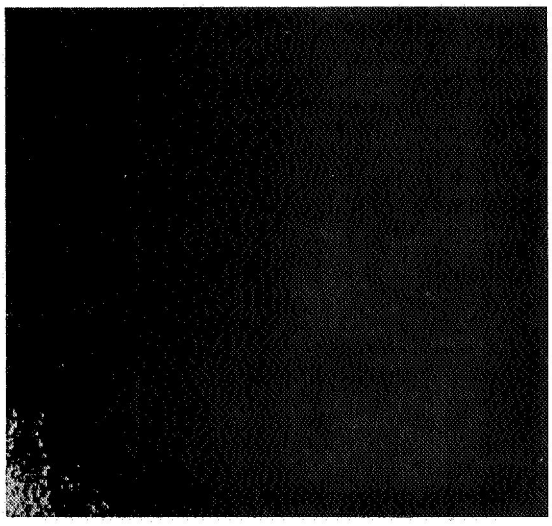
Figure 1-4. Fourier Transforms of Surveyor Spacecraft Boom



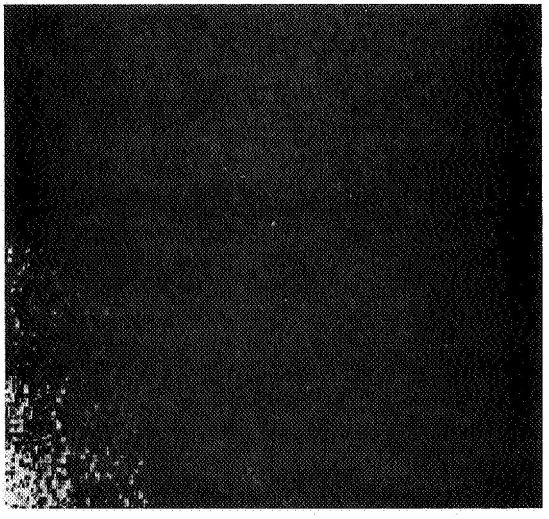
a) Original



b) Hadamard Transform of Hadamard Transform

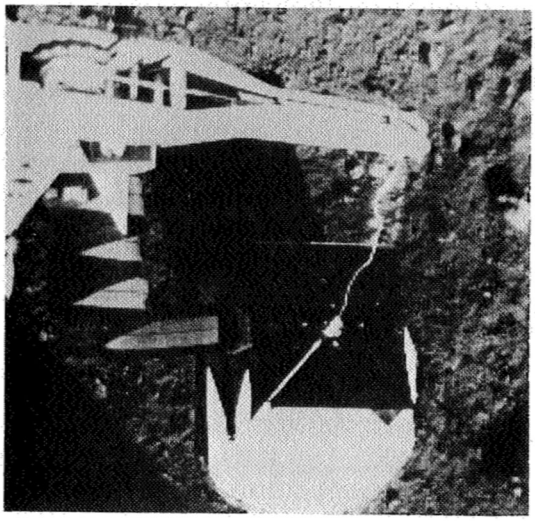


c) Magnitude of Hadamard Transform

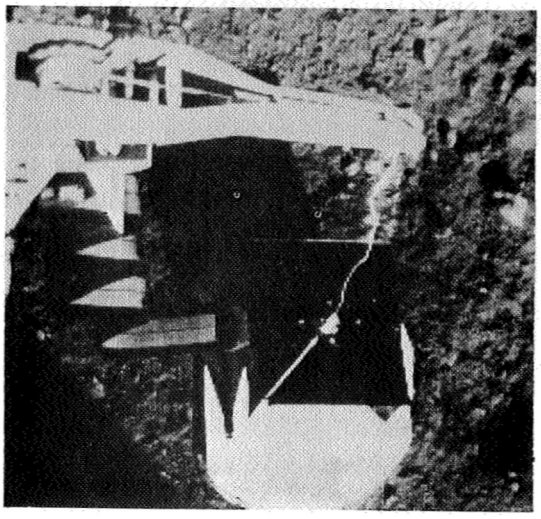


d) Logarithm of Magnitude of Hadamard Transform

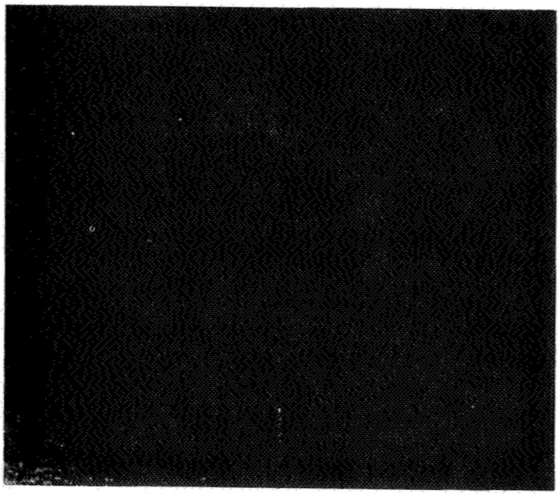
Figure 1-5. Hadamard Transforms of Surveyor Spacecraft Footpad



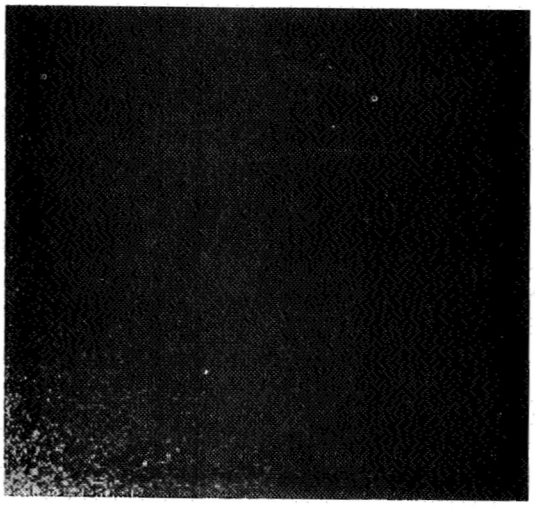
a) Original



b) Hadamard Transform of Hadamard Transform

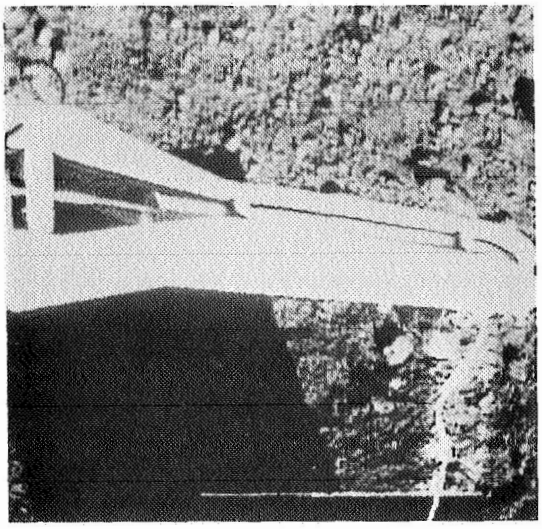


c) Magnitude of Hadamard Transform

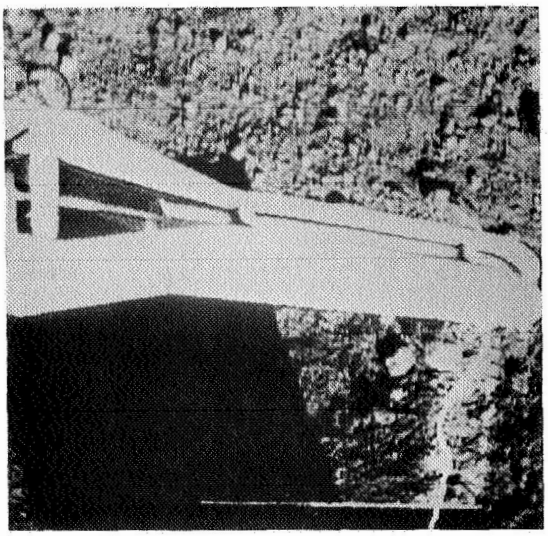


d) Logarithm of Magnitude of Hadamard Transform

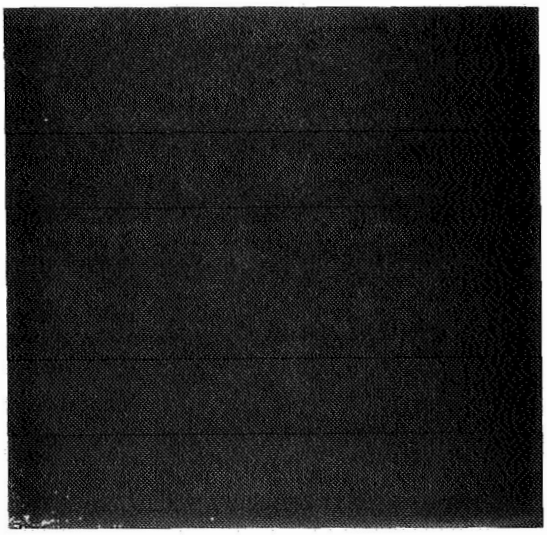
Figure 1-6. Hadamard Transforms of Surveyor Spacecraft Experimental Box



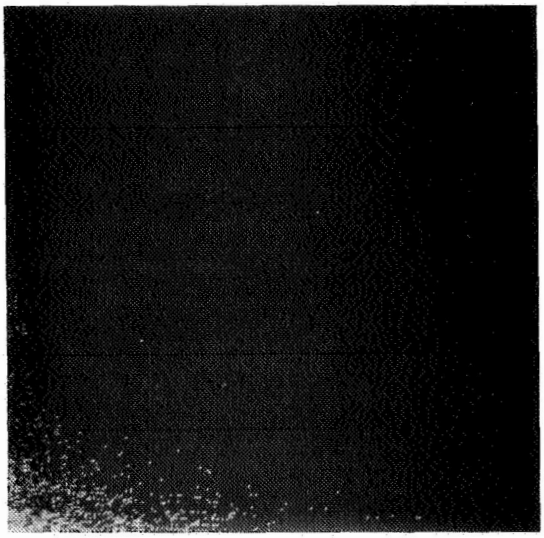
a) Original



b) Hadamard Transform of Hadamard Transform



c) Magnitude of Hadamard Transform



d) Logarithm of Magnitude of Hadamard Transform

Figure 1-7. Hadamard Transforms of Surveyor Spacecraft Boom

quantized to 64 grey levels have been transformed on a general purpose computer of the experimental image processing equipment described in Appendix C. Spatial and transform domains have been displayed on a cathode ray tube monitor for photographic recording. All transforms contain 256 by 256 sample points.

Figures 1-2 to 1-7 indicate that there is no apparent image degradation between the originals and the double transforms as a result of the image transform operations. The inherent correlation of elements in the originals has caused the energy in the transform domains to be squeezed toward the zero spatial frequencies. It is this characteristic of transformed images that is exploited to achieve a bandwidth reduction. The error immunity property of transform coding results from the inherent averaging operation of the transform. Each intensity sample of a reconstructed image is a weighted function of all transform samples. Hence, the magnitude of a single channel error is distributed over all of the reconstructed elements.

2. Image Transformation

Consideration is given in this section to the mathematical formulation of image transforms. The characteristics and properties of the Fourier, Hadamard, and other transforms are then developed, and finally conditions are given for the existence of fast computational algorithms.

2.1 Formulation

An original image may be represented by an array of intensity components or samples over the image surface by two dimensional sampling. In the transform coding system, it is conceptually possible to process the entire image or subsections of the image. The "best" image or subsection size is dependent upon the degree of spatial correlation of the image and the amount of processing permitted. For the present discussion an image array will be considered to be a square array of N^2 intensity samples described by the function $f(x, y)$, over the image coordinates (x, y). Then the two dimensional forward transform of the image array, $F(u, v)$, itself defined on a square array of N^2 points, may be expressed as

$$F(u,v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y)\, a(x,y,u,v) \qquad (2\text{-}1)$$

where $a(x, y, u, v)$ is the forward transformation kernel. The kernel is said to be separable if it can be written as

$$a(x,y,u,v) = a_1(x,u)\, a_2(y,v) \qquad (2\text{-}2)$$

A separable two dimensional transform can be computed in two steps. First, a one dimensional transform is taken along each row of the image, $f(x,y)$, yielding

$$F(u,y) = \sum_{x=0}^{N-1} f(x,y)\, a_1(x,u) \tag{2-3}$$

Next, a second one dimensional transform is taken along each column of $F(u,y)$ giving

$$F(u,v) = \sum_{y=0}^{N-1} F(u,y)\, a_2(y,v) \tag{2-4}$$

The transformation kernel is called separable symmetric if

$$a(x,y,u,v) = a_1(x,u)a_1(y,v) \tag{2-5}$$

For ease of implementation, the separable symmetric property is desirable. Furthermore, since the statistical intensity variations of most images are nearly the same in the vertical and horizontal directions only separable symmetric kernels usually need be considered.

A reverse transform may be defined as

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v)\, b(x,y,u,v) \tag{2-6}$$

where $b(x,y,u,v)$ is the reverse transformation kernel. For $f(x,y)$ and $F(u,v)$ to be transform pairs the following conditions are sufficient:

(a) $$a(x,y,u,v)\, b(x,y,u,v) = \frac{1}{N^2} \tag{2-7}$$

(b) $$a(x,y,u,v)\, b(\alpha,\beta,u,v) = \frac{1}{N} a(x-\alpha, y-\beta, u, v) = \frac{1}{N} b(\alpha-x, \beta-y, u, v) \quad (2\text{-}8)$$

(c) $$\sum_{u=0}^{N-1} \sum_{v=0}^{N-1} a(x-\alpha, y-\beta, u, v) = N\delta(x-\alpha, y-\beta) \quad (2\text{-}9)$$

It is often useful to express two dimensional transforms in matrix notation. For a transform kernel that is separable symmetric let:

$[f]$ = image matrix

$[F]$ = transformed image matrix

$[A]$ = transform matrix

Then by matrix multiplication

$$[F] = [A]\,[f]\,[A] \quad (2\text{-}10)$$

Now pre- and post-multiplication of each side of $[F]$ by a reverse transform matrix, $[B]$, gives

$$[\hat{f}] \equiv [B][F][B] = [B][A][f][A][B] \quad (2\text{-}11)$$

where $[\hat{f}]$ is, in general, an approximation of $[f]$. If the reverse transform matrix is the inverse matrix $[A]^{-1}$ of $[A]$, then

$$[\hat{f}] = [A]^{-1}[A][f][A][A]^{-1} \quad (2\text{-}12)$$

But

$$[A]^{-1}[A] = [A][A]^{-1} = [I] \quad (2\text{-}13)$$

where $[I]$ is the identity matrix. Hence

$$[\hat{f}] = [f] = [A]^{-1}[f][A]^{-1} \tag{2-14}$$

Thus, $f(x, y)$ and $F(u, v)$ can be expressed as two dimensional transform pairs if $[A]$ has an inverse.

If $[A]$ is a unitary matrix, then by definition

$$[A]^{-1} = [A^*]^T \tag{2-15}$$

where $[A^*]$ is the complex conjugate matrix of $[A]$ and where $[A]^T$ is the matrix transpose of $[A]$. A real, unitary matrix is called an orthogonal matrix. For such a matrix

$$[A]^{-1} = [A]^T \tag{2-16}$$

Finally, if $[A]$ is a symmetric orthogonal matrix then

$$[A]^{-1} = [A] \tag{2-17}$$

If the forward transformation matrix, $[A]$, is constrained to be orthogonal, then the transformation can be interpreted as a decomposition of the image data into a generalized two dimensional spectrum. Each spectral component in the transform domain corresponds to the amount of energy of that spectral orthogonal function within the original image. In this context the concept of frequency may now be generalized to include transformations of orthogonal functions other than sine and cosine waveforms. This type of generalized spectral analysis is useful in the investigation of specific orthogonal decompositions which are best suited for particular classes of images.

The following subsections contain an analysis of the Fourier, Hadamard, Kronecker, and Eigenvector transformations with particular emphasis on their applicability to image coding.

2.2 Fourier Transform

The discrete Fourier transform, with and without efficient computational algorithms, has long been used for signal analysis [6]. Only recently have Fourier transform methods been utilized for image coding [9-12]. The two dimensional Fourier transform of an image field, f(x, y), may be expressed as

$$F(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) \exp\left\{ -\frac{2\pi i}{N} (ux + yv) \right\} \qquad (2\text{-}18)$$

The inverse Fourier transform which reconstructs the original image is given by

$$f(x,y) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{y=0}^{N-1} F(u,v) \exp\left\{ \frac{2\pi i}{N} (ux + vy) \right\} \qquad (2\text{-}19)$$

Since the transform kernels are separable and symmetric the two dimensional transform can be computed as two sequential one dimensional transforms.

The terms u and v are called the spatial frequencies of the image in analogy with time series analysis. When the Fourier transform relationship is expressed in the form given by equation (2-18), the origin, or zero spatial frequency term appears in the corner of the

transform plane. For display purposes it is convenient to shift the origin to the center of the transform domain. This is easily accomplished by multiplying the image by the function $(-1)^{x+y}$ before the transformation. Let

$$G(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} (-1)^{x+y} f(x,y) \exp\left\{ -\frac{2\pi i}{N} (ux + vy) \right\} \tag{2-20}$$

But since

$$(-1)^{x+y} = e^{i\pi(x+y)} \tag{2-21}$$

the function $G(u,v)$ may be written as

$$G(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \exp\left\{ -\frac{2\pi i}{N} \left[\left(u - \frac{N}{2}\right) + \left(v - \frac{N}{2}\right) y \right] \right\} \tag{2-22}$$

or

$$G(u,v) = F\left(u - \frac{N}{2}, v - \frac{N}{2}\right) \tag{2-23}$$

Thus, the origin moves to the center of the transform domain.

Even though $f(x,y)$ is a real positive function, its transform, $F(u,v)$, is in general complex. Thus, while the image contains N^2 components, the transform contains $2N^2$ components, the real and imaginary or magnitude and phase components of each spatial frequency. However, since $f(x,y)$ is a real positive function, $F(u,v)$ exhibits a property of conjugate symmetry. To illustrate this property let

$$F(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \left\{ \cos \frac{2\pi}{N} (ux+vy) - i \sin \frac{2\pi}{N} (ux+vy) \right\} \quad (2\text{-}24)$$

The Fourier transform can be divided into real and imaginary components as

$$F(u,v) = F_R(u,v) - i F_I(u,v) \quad (2\text{-}25)$$

where, since $f(x,y)$ is real,

$$F_R(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos \frac{2\pi}{N} (ux+vy) \quad (2\text{-}26)$$

and

$$F_I(u,v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \sin \frac{2\pi}{N} (ux+vy) \quad (2\text{-}27)$$

The cosine is even in u and v, and the sine is odd in u and v, hence

$$F_R(u,v) = F_R(-u,-v) \quad (2\text{-}28)$$

and

$$F_I(u,v) = -F_I(-u,-v) \quad (2\text{-}29)$$

Consequently $F(u,v) = F^*(-u,-v)$ (2-30)

Figure 2-1 illustrates the conjugate symmetry property of the Fourier transform when the zero spatial frequency term is located at the center of the transform plane. Samples in quadrants (1) and (3) are complex conjugates of one another as are samples in quadrants (2) and (4). This property is further illustrated by magnitude displays of the Fourier transforms shown in figures 1-2 through 1-4. As a result of the conjugate symmetry property of the Fourier transform it is only necessary to

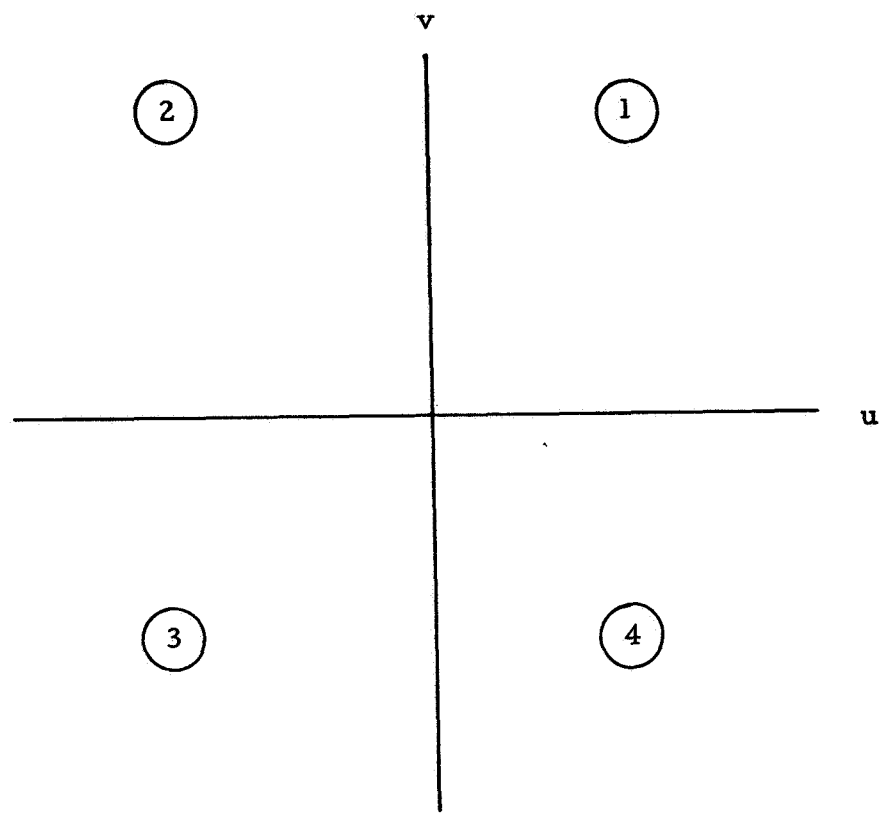


Figure 2-1. Fourier Domain Quadrants

transmit the samples of one half of the transform plane; the other half can be reconstructed from the half plane samples transmitted*. Hence, the Fourier transform of an image can be described by N^2 data components.

The two dimensional Fourier transform of an image is essentially a Fourier series representation of a two dimensional field. For the Fourier series representation to be valid the field must be periodic. Thus, the original image must be considered to be periodic horizontally and vertically as shown in figure 2-2. The right side of the image therefore abuts the left side and the top and bottom of the image are adjacent. Spatial frequencies along the coordinate axes of the transform plane arise from these transitions. Although these are false spatial frequencies from the standpoint of being necessary for representing the image within the image boundary, they do not impair reconstruction. On the contrary, these spatial frequencies are required to reconstruct the sharp boundaries of the image.

The Fourier transform can be easily expressed in a matrix formulation by letting

$$\mathcal{W} \equiv \exp\left\{-\frac{2\pi i}{N}\right\} \tag{2-31}$$

Then

$$[F] = [A][f][A] \tag{2-32}$$

*A reconstruction of the original can be obtained from the half plane transform samples directly by a Hilbert filtering technique [9].

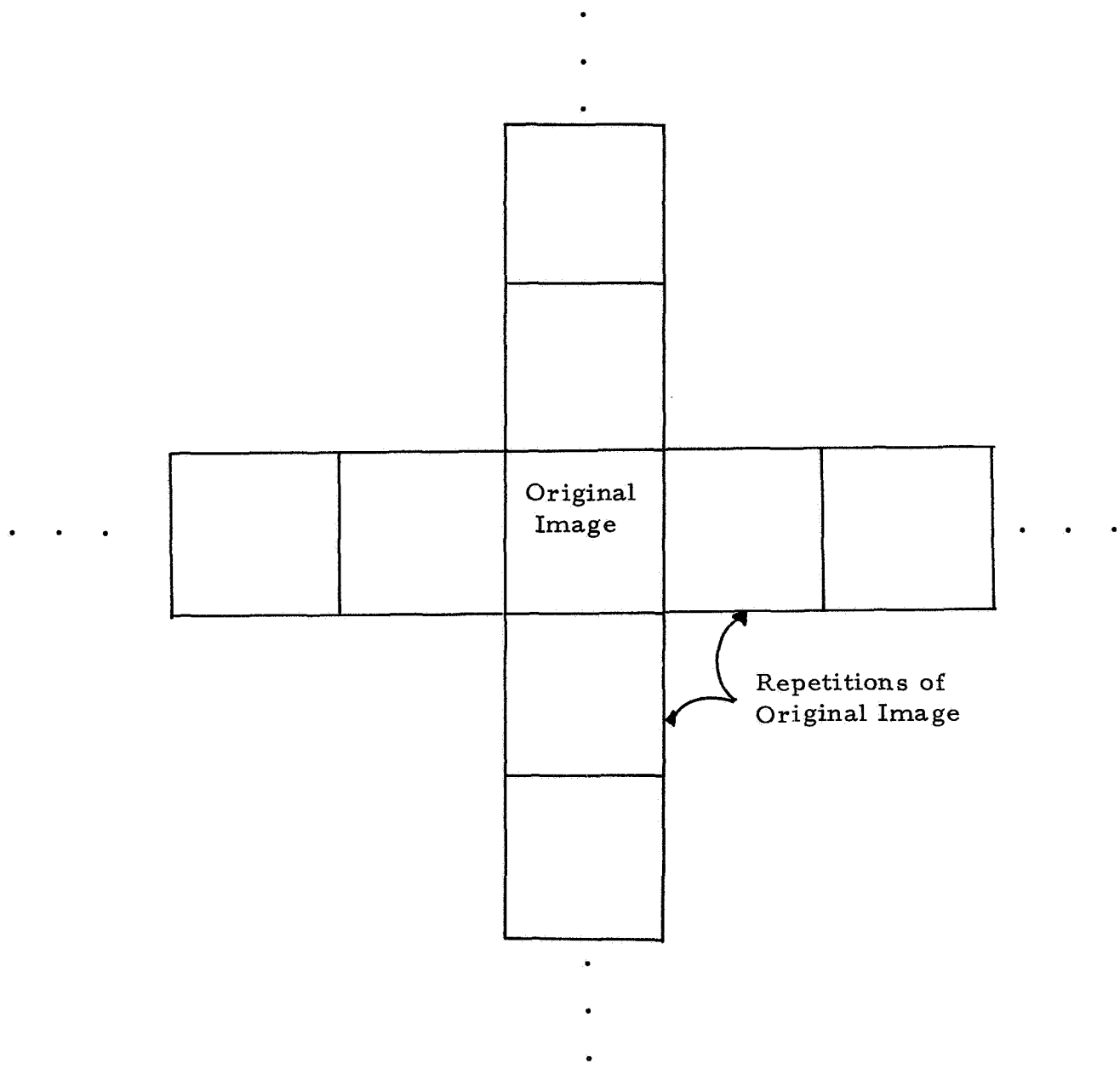


Figure 2-2. Fourier Series Representation of an Image

where

$$[A] = \frac{1}{\sqrt{N}} \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ \vdots \\ N-1 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & \cdots & N-1 \end{array} \\ \begin{bmatrix} W^0 & W^0 & W^0 & W^0 & \cdots & W^0 \\ W^0 & W^1 & W^2 & W^3 & \cdots & W^{N-1} \\ W^0 & W^2 & W^4 & W^6 & \cdots & W^{2(N-1)} \\ W^0 & W^3 & & & & \\ \vdots & \vdots & & & & \\ W & W^{N-1} & \cdots & \cdots & \cdots & W^{(N-1)^2} \end{bmatrix} \\ x \rightarrow \end{array} \begin{array}{l} \\ \\ \\ \\ \\ \\ u \\ \downarrow \end{array} \qquad (2\text{-}33)$$

The matrix is obviously symmetric. Computational simplification can be realized since

$$W^{ux} = W^{ux \bmod N} \qquad (2\text{-}34)$$

2.3 Hadamard Transform

The Hadamard transform is based upon the Hadamard matrix which is a square array of plus and minus ones whose rows and columns are orthogonal to one another [16-18]. If [H] is an N by N Hadamard matrix, then the product of N and its transpose is

$$[H][H]^T = N[I] \qquad (2\text{-}35)$$

If H is a symmetric Hadamard matrix, then equation (2-35) reduces to

$$[H][H] = N[I] \qquad (2\text{-}36)$$

A Hadamard matrix multiplied by the normalization factor $\frac{1}{\sqrt{N}}$ is an orthonormal matrix.

The lowest order Hadamard matrix is the Hadamard matrix

$$[H_2] = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2-37}$$

It is known that if a Hadamard matrix of order N exists (N >2), then $N \equiv 0 \pmod 4$. The existence of a Hadamard matrix for every value of N satisfying this requirement has not been shown, but constructions are available for nearly all permissible values of N up to 200. The simplest construction is for a Hadamard matrix of order $N = 2^n$ where n is an integer. In this case if $[H_N]$ is a Hadamard matrix of order N, the matrix

$$[H_{2N}] = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix} \tag{2-38}$$

is a Hadamard matrix of order 2N. Figure 2-3 contains several Hadamard matrices of order $N = 2^n$. Another simple construction is possible if $[G_M]$ and $[H_N]$ are Hadamard matrices of orders M and N, respectively. Then there exists a Hadamard matrix of order M • N given by

$$H_{M\cdot N} = \begin{bmatrix} g_{11}H_N & g_{12}H_N \cdots & g_{1M}H_N \\ g_{21}H_N & & \vdots \\ \vdots & & \vdots \\ g_{M1}H_N & \cdots & g_{MM}H_N \end{bmatrix} \tag{2-39}$$

	Matrix	Sequency
N = 2	$\begin{bmatrix} + & + \\ + & - \end{bmatrix}$	0 1

	Matrix	Sequency
N = 4	$\begin{bmatrix} + & + & + & + \\ + & - & + & - \\ + & + & - & - \\ + & - & - & + \end{bmatrix}$	0 3 1 2

	Matrix	Sequency
N = 8	$\begin{bmatrix} + & + & + & + & + & + & + & + \\ + & - & + & - & + & - & + & - \\ + & + & - & - & + & + & - & - \\ + & - & - & + & + & - & - & + \\ + & + & + & + & - & - & - & - \\ + & - & + & - & - & + & - & + \\ + & + & - & - & - & - & + & + \\ + & - & - & + & - & + & + & - \end{bmatrix}$	0 7 3 4 1 6 2 5

Figure 2-3. Hadamard matrices of order $N = 2^n$

Other constructions are given in references [19 to 21]. The set of known Hadamard matrices is sufficiently numerous to satisfy almost all size requirements for image coding.

A frequency interpretation can be given to the Hadamard matrix generated from the core matrix of equation (2-37). Along each row of the Hadamard matrix the frequency is called the number of changes in sign. Harmuth has coined the word "sequency" to designate the number of sign changes [22]. Figure 2-3 gives the sequency interpretation for several Hadamard matrices of binary order. It is possible to construct a Hadamard matrix of order $N = 2^n$ that has frequency components at every integer from 0 to N-1.

This frequency interpretation of the rows of a Hadamard matrix leads one to consider the rows to be equivalent to rectangular waves ranging between ± 1 with a sub-period of $\frac{1}{N}$ units. Such functions are called Walsh functions [23-27] and are further related to the Rademacher functions [28]. Thus, in this context the Hadamard matrix merely performs the decomposition of a function by a set of rectangular waveforms rather than the sine-cosine waveforms associated with the Fourier transform.

For symmetric Hadamard matrices of order $N = 2^n$, the two dimensional Hadamard transform may be written in series form as

$$F(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)(-1)^{p(x,y,u,v)} \qquad (2\text{-}40)$$

where $p(x, y, u, v) \equiv \sum_{i=0}^{N-1} (u_i x_i + v_i y_i)$. The terms u_i, v_i, x_i, and y_i are the binary representations of u, v, x, and y respectively. For example

$$(u)_{DECIMAL} = (u_{n-1} u_{n-2} \cdots u_1 u_0)_{BINARY} \tag{2-41}$$

where $u_i \in \{0, 1\}$. In equation (2-40) the summation in the exponent is performed modulo two. This representation of the Hadamard transform is for the Hadamard matrix in "natural" form as given by equation (2-38). Another series representation exists for a Hadamard matrix in "ordered" form in which the sequency of each row is larger than the preceding row. By this representation

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y)(-1)^{q(x, y, u, v)} \tag{2-42}$$

where

$$q(x, y, u, v) \equiv \sum_{i=0}^{n-1} [g_i(u) x_i + g_i(v) y_i] \tag{2-43}$$

and

$$\begin{aligned} g_0(u) &\equiv u_{n-1} \\ g_1(u) &\equiv u_{n-1} + u_{n-2} \\ g_2(u) &\equiv u_{n-2} + u_{n-3} \\ &\vdots \\ g_{n-1}(u) &\equiv u_1 + u_0 \end{aligned} \tag{2-44}$$

The two dimensional Hadamard transform may be computed in either natural or ordered form with an algorithm analogous to the fast

Fourier transform computer algorithm.

2.4 Kronecker Matrix Transforms

The Kronecker matrix transforms are a generalized class of mathematical transformations that are amenable to highly efficient computer implementation. Some of these matrices may be useful for image coding applications.

Consider the class of matrices formed by the kronecker product operation. Let the core matrix be square and of dimension p by p with entries $m_{0,i,j}$ where i and j range from zero through p-1.

$$[H_1] = \begin{bmatrix} m_{0,0,0} & m_{0,0,1} & \cdots & m_{0,0,p-1} \\ m_{0,1,0} & & & \\ \vdots & & & \\ m_{0,p-1,0} & \cdots & & m_{0,p-1,p-1} \end{bmatrix} \qquad (2\text{-}45)$$

Here the first index represents the class of entries corresponding to a particular dimension in the kronecker product operation. For example, the p^2 by p^2 matrix, $[H_2]$, formed by the kronecker operation can be expressed as

$$\left[H_2\right]=\begin{bmatrix} m_{1,0,0}H_1 & m_{1,0,1}H_1 & \cdots & m_{1,0,p-1}H_1 \\ m_{1,1,0}H_1 & m_{1,1,1}H_1 & \cdots & m_{1,1,p-1}H_1 \\ \vdots & & & \\ m_{1,p-1,0}H_1 & \cdots & \cdots & m_{1,p-1,p-1}H_1 \end{bmatrix} \tag{2-46}$$

where the class of entries $m_{1,i,j}$ form the coefficients of the $\left[H_1\right]$ submatrices in the $\left[H_2\right]$ matrix. In general, the class of entries $m_{n-1,i,j}$ form the coefficients of the $\left[H_{n-1}\right]$ submatrices in the $\left[H_n\right]$ matrix. Thus,

$$\left[H_n\right]=\begin{bmatrix} m_{n-1,0,0}H_{n-1} & m_{n-1,0,1}H_{n-1} & \cdots & m_{n-1,0,p-1}H_{n-1} \\ m_{n-1,1,0}H_{n-1} & & & m_{n-1,1,p-1}H_{n-1} \\ \vdots & & & \\ m_{n-1,p-1,0}H_{n-1} & & & m_{n-1,p-1,p-1}H_{n-1} \end{bmatrix} \tag{2-47}$$

where $\left[H_n\right]$ is a p^n by p^n matrix.

When operating with kronecker matrices within a computer, it becomes desirable to store a representation (algorithm) of the entries of the matrix rather than the matrix itself. Towards this end, consider the locations in the matrix to be described by their lexicographic or dictionary sequence representation. In other words, a given index of matrix $\left[H_n\right]$ can be represented by n digits each of which can take on the value zero through p-1. Representing the horizontal index by u and the

vertical index by x, the names of the rows and columns in dictionary sequence for the $[H_2]$ matrix with p = 3 are

$$[H_2] = \begin{array}{c} \\ x \downarrow \end{array} \begin{array}{c} u \longrightarrow \\ \begin{array}{cccccccccc} & 00 & 01 & 02 & 10 & 11 & 12 & 20 & 21 & 22 \end{array} \\ \begin{array}{c} 00 \\ 01 \\ 02 \\ 10 \\ 11 \\ 12 \\ 20 \\ 21 \\ 22 \end{array} \begin{bmatrix} & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & H_2(x,u) & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \end{bmatrix} \end{array} \qquad (2\text{-}48)$$

Representing the u and x variables in the dictionary number system mod p requires n digits to allow u and x to range over zero to p^n. Therefore u and x can be described by

$$u = u_{n-1}\, u_{n-2} \cdots u_1 u_0 \qquad u_i \in \{0, 1, \ldots, p-1\} \qquad (2\text{-}49a)$$

$$x = x_{n-1}\, x_{n-2} \cdots x_1 x_0 \qquad x_i \in \{0, 1, \ldots, p-1\} \qquad (2\text{-}49b)$$

Using such a notation allows the entires of the p by p core matrix $[H_1]$ given by equation (2-45) to be described by the equation

$$[H_1(x,u)] = \prod_{i=0}^{p-1} \prod_{j=0}^{p-1} m_{0,i,j}^{\delta(x_0 - i)\, \delta(u_0 - j)} \qquad (2\text{-}50)$$

where $\delta(a-b)$ is the delta function which takes on the value "one" whenever

a = b and zero otherwise. The representation of equation (2-50) can be interpreted as multiplying all entries of the core matrix, equation (2-45), together and noting that all but one entry will be raised to the zero power. The entries of the p^2 by p^2 matrix, $\left[H_2\right]$, equation (2-46), can now be represented as

$$\left[H_2(x,u)\right] = \prod_{i=0}^{p-1}\prod_{j=0}^{p-1} m_{1,i,j}^{\delta(x_1-i)\delta(u_1-j)} \prod_{i=0}^{p-1}\prod_{j=0}^{p-1} m_{0,i,j}^{\delta(x_0-i)\delta(u_0-j)} \tag{2-51}$$

where, again, the exponents determine the correct product of entries for a given u and x. In general, the entries for $\left[H_n\right]$ can be represented as

$$\left[H_m(x,u)\right] = \prod_{r=0}^{n-1}\prod_{i=0}^{p-1}\prod_{j=0}^{p-1} m_{r,i,j}^{\delta(x_r-i)\delta(u_r-j)} \tag{2-52}$$

following the recursive notation of equation (2-50) and (2-51). Representation of the rows or columns of a kronecker matrix in the form of equation (2-52) now allows the generation of any single element, column, or row of the matrix without storage of the entire matrix array. This becomes particularly important for large matrices.

In addition to representing the kronecker matrices in closed product form, it is important to also point out that vector multiplication with the above described matrices can be implemented on the order of $pN \log_p N$ operations where $N = p^n$ is the dimension of the $\left[H_n\right]$ kronecker matrix. This should be contrasted with the N^2 operations normally required. This result was pointed out by Good [29] and leads to the fast Fourier transform algorithm [4] as well as the fast Hadamard

transform algorithm [13,30]. The class of kronecker matrices described above can be decomposed into a product of matrices each of which has only p entires in a given row or column. Thus for the $[H_n]$ kronecker matrix of equation (2-47) there exist n matrices, each of dimension p^n, such that when multiplied together, they will equal $[H_n]$. These matrices can be described as

$$[G_r] = \begin{bmatrix} m_{r,0,0} \cdots m_{r,0,p-1} & & & \\ & m_{r,0,0} \cdots m_{r,0,p-1} & & \\ & & \vdots & \\ & & m_{r,0,0} \cdots m_{r,0,p-1} & \\ m_{r,1,0} \cdots m_{r,1,p-1} & & & \\ & m_{r,1,0} \cdots m_{r,1,p-1} & & \\ & & \vdots & \\ & & m_{r,1,0} \cdots m_{r,1,p-1} & \\ \cdots\cdots\cdots & \cdots\cdots\cdots & \cdots\cdots\cdots & \\ m_{r,p-1,0} \cdots m_{r,p-1,p-1} & & & \\ & m_{r,p-1,0} \cdots m_{r,p-1,p-1} & & \\ & & \vdots & \\ & & m_{r,p-1,0} \cdots m_{r,p-1,p-1} & \end{bmatrix} \tag{2-53}$$

Then

$$[H_n] = [G_{n-1}][G_{n-2}] \cdots [G_1][G_0] \tag{2-54}$$

Now if a vector is multiplied by $[H_n]$, N^2 operations will be required

whereas if the vector is multiplied by $\left[G_{n-1}\right]$, pN operations will be required. If the resulting vector is multiplied by $\left[G_{n-2}\right]$, another pN operations will be required. If this step is carried out $n = \log_p N$ times, then a total of $p N \log_p N$ operations are necessary.

By using the Good algorithm described above and by using the closed product representation of equation (2-52) a kronecker matrix of large dimension can be generated and matrix-manipulated without storing the N^2 term matrix. Conceivably the set of coefficients $m_{r,i,j}$ of equation (2-52) could all be distinct, in which case a total of $(p^2)^n = N^2$ coefficients must be stored. However, when the class of $m_{r,i,j}$ are not all distinct, considerable savings can be achieved. It is instructive to investigate the class of matrices generated by the kronecker operation with the $m_{r,i,j} = m_{s,i,j}$ for all r and s. In such a situation equation (2-52) reduces to

$$\left[H_n(x,u)\right] = \prod_{i=0}^{p-1}\prod_{j=0}^{p-1} m_{i,j}^{\sum_{r=0}^{n-1}\delta(x_r-i)\delta(u_r-j)} \tag{2-55}$$

Now only p^2 coefficients need be stored compared to $p^{2n} = N^2$ terms.

If the general matrices described above are generated from a two by two core matrix, the closed product representation analogous to equation (2-52) becomes particularly convenient to implement. Let the core matrix $\left[H_1\right]$ be

$$\left[H_1\right] = \begin{bmatrix} A_0 & B_0 \\ C_0 & D_0 \end{bmatrix} \tag{2-56}$$

and $\left[H_n\right]$ can be represented as

$$\left[H_n\right] = \begin{bmatrix} A_{n-1} & H_{n-1} & B_{n-1} & H_{n-1} \\ C_{n-1} & H_{n-1} & D_{n-1} & H_{n-1} \end{bmatrix} \tag{2-57}$$

The closed form product representation now becomes

$$\left[H_n(x,u)\right] = \prod_{r=0}^{n-1} A_r^{\bar{x}_r \bar{u}_r} B_r^{\bar{x}_r u_r} C_r^{x_r \bar{u}_r} D_r^{x_r u_r} \tag{2-58}$$

where the exponent operations become Boolean "and" operations, and the bar over the binary variable represents the complement value. For the case in which $A_r = A_s$, $B_r = B_s$, $C_r = C_s$, $D_r = D_s$ for all r and s, the representation again simplifies and becomes

$$\left[H_n(x,u)\right] = A^{\sum_{r=0}^{n-1} \bar{u}_r \bar{x}_r} B^{\sum_{r=0}^{n-1} \bar{x}_r u_r} C^{\sum_{r=0}^{n-1} x_r \bar{u}_r} D^{\sum_{r=0}^{n-1} x_r u_r} \tag{2-59}$$

Equation (2-59) is particularly suited for special purpose digital implementation as the exponent operations require simply counting the number of "ones" obtained from a parallel component wise register "and" operation on the values of u and x. This means that generation of the rows, columns, or specific elements of the matrix $\left[H_n\right]$ requires storage of only four variables (A, B, C, D) and simple register "and" operations for any dimension $N = 2^n$. Then to implement a vector-matrix product will require $2Nn = 2N \log_2 N$ operations with a storage requirement of only 4 variables.

Orthogonal matrix transformations are particularly desirable

for preserving inner products as well as describing a generalized spectral analysis transform domain. A general class of orthogonal transforms of particular interest can be obtained from the kronecker matrices described in earlier sections by requiring that the sets of variables $\{m_{r,0,0}, \ldots, m_{r,p-1,p-1}\}$ satisfy the orthogonality requirement for all $r = 0, \ldots, n-1$. If this constraint is satisfied, then the matrix, $\left[H_n\right]$ of equation (2-47) or (2-52) becomes orthogonal and is a valid candidate for a kernel in a generalized spectral decomposition problem. For the power of two case the orthogonality constraint on the sets $\{A_r, B_r, C_r, D_r\}$ for all $r = 0, \ldots, n-1$ reduces to

$$A_r^2 + B_r^2 = 1 \qquad (2\text{-}60a)$$

$$C_r^2 + D_r^2 = 1 \qquad (2\text{-}60b)$$

$$A_r C_r + B_r D_r = 0 \qquad (2\text{-}60c)$$

for each r. In this case equation (2-58) becomes the kernel of the transform and when the sets $\{A_r, B_r, C_r, D_r\}$ are all identical, equation (2-59) becomes the kernel.

If it is desired to make the kernel matrix symmetric so that a transformation taken twice results in the original function again, then further simplifications result in the closed form representation of the matrix $\left[H_n\right]$. The requirement for symmetry and orthogonality for the case of identical sets $\{A_r, B_r, C_r, D_r\}$ for all r is

$$B = C \qquad (2\text{-}61a)$$

$$A^2 + B^2 = 1 \qquad (2\text{-}61b)$$

$$B^2 + D^2 = 1 \tag{2-62b}$$

$$(A + D)B = 0 \tag{2-63b}$$

The entries of the matrix $\left[H_n\right]$ then become

$$\left[H_n(x,u)\right] = A^{\sum_{r=0}^{n-1} \bar{u}_r \bar{x}_r} \; B^{\sum_{r=0}^{n-1} u_r \oplus x_r} \; D^{\sum_{r=0}^{n-1} u_r x_r} \tag{2-64}$$

where $\oplus$ implies an exclusive "or" Boolean operation. Notice that the exponents can be determined by summing the result of parallel register operations (Boolean "and" and Boolean "exclusive or") on the variables u and x. However, equation (2-64), under the constraint that B = 0 satisfying equation (2-63b), reduces to

$$\left[H_n(x,u)\right] = A^{\sum_{r=0}^{n-1} \bar{u}_r} \; D^{\sum_{r=0}^{n-1} u_r} \; \delta(u-x) \tag{2-65a}$$

or

$$\left[H_n(x,u)\right] = \left(\frac{D}{A}\right)^{\sum_{r=0}^{n-1} u_r} \delta(u-x) \tag{2-65b}$$

which is a diagonal matrix. The alternative constraint to equation (2-63b) is that A = -D in which case more interesting orthogonal symmetric matrices result. Thus,

$$\left[H_n(x,u)\right] = A^{\sum_{r=0}^{n-1} u_r \odot x_r} \; B^{\sum_{r=0}^{n-1} u_r \oplus x_r} \; (-1)^{\sum_{r=0}^{n-1} u_r x_r} \tag{2-66a}$$

or

$$\left[H_n(x,u)\right] = A^n\left(\frac{D}{A}\right)^{\sum_{r=0}^{n-1} u_r \oplus x_r} (-1)^{\sum_{r=0}^{n-1} u_r x_r} \qquad (2\text{-}66b)$$

where $\odot$ is the Boolean "coincidence" operation equivalent to the complement of the exclusive "or" operation. The class of orthogonal matrices described by equations (2-66a) or (2-66b) are a two parameter family of sets of kronecker matrices subject to the constraint that $A^2 + B^2 = 1$. Consequently, valid 2-tuples satisfying this requirement are $\{\cos\theta, \sin\theta\}$, $\{1,0\}$, $\{3/5, 4/5\}$, $\{1/\sqrt{2}, 1/\sqrt{2}\}$ and many others. Further study is required to determine the applicability of kronecker matrix transforms to image coding.

2.5 Karhunen-Loeve Transform

In the transform threshold sampling technique of bandwidth reduction, presented in Section 5, only those transform samples whose magnitudes are greater than a threshold level are coded. The optimum transform for minimizing the number of transform samples lying above the threshold while satisfying a mean square error criterion between the original and the reconstructed image is the Karhunen-Loeve transform [31,32,45,48-52]. This transform is composed of Eigenvectors of the correlation matrix of the original image, or class of images, to be coded.

The Karhunen-Loeve transform is not, in general, separable. Hence, the original image must be regarded as a vector rather than a matrix. Let

$$[f(z)] \equiv [f(x,y_1), f(x,y_2), \ldots, f(x,y_N)] \qquad (2\text{-}67)$$

be a row vector composed of lines of the original image examined in the normal "raster" pattern. The correlation matrix of the image is an N^2 by N^2 matrix of the form*

$$[R] = E\{z_i z_j\} \qquad i=1,2,\ldots,N;\ j=1,2,\ldots,N \tag{2-68}$$

If the correlation matrix is not known, it can be estimated from an ensemble of original images. Let $[f_k(z)]$ represent the kth of n images from which $[R]$ is to be estimated. Then

$$[R] \approx \frac{1}{n} \sum_{k=1}^{n} [f_k(z)]^T [f_k(z)] \tag{2-69}$$

The forward Karhunen-Loeve transform is the orthogonal matrix composed of the Eigenvectors of the correlation matrix arranged such that

$$[A][R][A] = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_{N^2} \end{bmatrix} \tag{2-70}$$

where $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{N^2}$ are the Eigenvalues of $[R]$ arranged in descending order. The Karhunen-Loeve transform, F(w), of the original image is then

$$[F(w)] = [f(z)][A] \tag{2-71}$$

And the reverse transform is

*$E\{\cdot\}$ = expected or mean value of the function within the brackets.

$$[B] = [A]^T \tag{2-72}$$

If only the first M of the N^2 columns of [A] are employed in the transform, i.e.,

$$\underbrace{[F_M(w)]}_{\substack{1 \times M \\ \text{matrix}}} = \underbrace{[f(z)]}_{\substack{1 \times N^2 \\ \text{matrix}}} \underbrace{[A_M]}_{\substack{N^2 \times M \\ \text{matrix}}} \tag{2-73}$$

then the mean square error, ε, is

$$\varepsilon = \sum_{k=M+1}^{N^2} \lambda_k \tag{2-74}$$

Since the λ_k are monotonically decreasing in value the error will be minimum for any M.

There are two major problems associated with the use of the Karhunen-Loeve transform for image coding. The first is that a great amount of computation must be performed. The correlation matrix must be estimated if it is not known. Next the correlation matrix must be diagonalized to determine its Eigenvalues and Eigenvectors. Finally, the transform itself must be taken. In general, there is no fast computational algorithm for the transform. The second difficulty with the Karhunen-Loeve transform is that the mean square error is not a valid error criterion for many types of images.

However, for those classes of images for which the mean square error criterion is valid, the Karhunen-Loeve transformation may find application as a standard for bandwidth reduction capability. Furthermore,

if the image is broken up into smaller subsections the computation of the transform may prove feasible.

2.6 Computational Algorithms

A characteristic of great importance for an image transform is the existence of a fast computational algorithm of the type available for the Fourier and Hadamard transforms. For a fast algorithm to exist it is necessary that the transform be factorable into matrices containing many zero elements [33].

As an example of matrix factorization, consider the Hadamard transform of order N = 8. It can be factored as follows.

$$\left[H_8\right] = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & - & 1 & - & 1 & - & 1 & - \\ 1 & 1 & - & - & 1 & 1 & - & - \\ 1 & - & - & 1 & 1 & - & - & 1 \\ 1 & 1 & 1 & 1 & - & - & - & - \\ 1 & - & 1 & - & - & 1 & - & 1 \\ 1 & 1 & - & - & - & - & 1 & 1 \\ 1 & - & - & 1 & - & 1 & 1 & - \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & - & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & - & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & - & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & - \end{bmatrix}^3 \qquad (2\text{-}75)$$

Multiplication of a row vector of length 8 by the matrix on the left in equation (2-75) requires 8(8-1) = 56 addition operations. Mutliplication of the vector successively by each of the factors of $\left[H_8\right]$ requires only one addition per column per factor for each factor, or a total of (1)(8)(3) = 24 additions.

It is not possible to identify matrices that are factorable into

matrices containing many zero elements. No algorithms exist for determining the best factorization of factorable matrices. Therefore, the only recourse in finding an efficient computational algorithm for an arbitrary transform matrix is to generate trial factorizations and compare them.

Fast computational algorithms for the Hadamard and Fourier transforms are presented in Appendicies A and B respectively.

3. Analysis of Image Transforms

The development of efficient quantization and coding methods for image transforms requires an understanding of the statistical distribution of energy in the transform domain and bounds on the distribution of the energy. In this section a stochastic model of transform samples is developed, energy bounds are derived, and computational requirements to preserve the energy distribution are presented.

3.1 Statistical Analysis

A complete statistical description of the effects of a general transformation operator on an original image is not possible. However, considerable insight into the general statistical description can be obtained from the Fourier transform by the relation between the Fourier transform spatial frequency and the concept of the generalized frequency of a transform.

In the statistical analysis of the two dimensional Fourier transform let $\emptyset(x', y')$ be a continuous two dimensional wide sense stationary random process with a bounded and continuous power spectral density, $D(u, v)$, where u and v are real. It is desired to observe the process over the two dimensional window, $(-I, -I; I, I)$, and to sample the process at N^2 uniformly spaced points within the window of observation. A new process, $F_{N,I}(u, v)$ depending on both the window of observation and the sampling period within the window, is formed as follows

$$F_{N,I}(u,v) = \frac{1}{N} \sum_{x,y=0}^{N-1\,N-1} \phi\left(\frac{xI}{N}, \frac{yI}{N}\right) \exp\left\{\frac{2\pi iI}{N}(ux+vy)\right\} \qquad (3\text{-}1)$$

The variance of $F_{N,I}(u,v)$ may be expressed in terms of a covariance function, ρ, on the process, ϕ. Thus

$$\sigma^2_{F_{N,I}}(u,v) = \frac{1}{N^2} \sum_{\tau,T=0}^{N-1\,N-1} \epsilon_\tau \epsilon_T (n-\tau)(N-T)\rho\left(\frac{\tau I}{N}, \frac{TI}{N}\right) \cos\frac{2\pi I}{N}(u\tau + vT) \qquad (3\text{-}2)$$

where τ and T are integer values representing the two dimensional shift in the sampled function $\phi\left(\frac{xI}{N}, \frac{yI}{N}\right)$ with itself. The terms ϵ_τ and ϵ_T are Neumann factors taking the values $\epsilon_0 = 1$ and $\epsilon_\tau = 2$ for all $\tau \neq 0$. Equation (3-2) can be expressed as

$$\sigma^2_{F_{N,I}}(u,v) = \sum_{\tau,T=0}^{N-1\,N-1} \epsilon_\tau \epsilon_T \left(1-\frac{\tau I}{N}\right)\left(1-\frac{TI}{NI}\right)\rho\left(\frac{\tau I}{N}, \frac{TI}{N}\right) \cos\frac{2\pi I}{N}(u\tau + vT) \qquad (3\text{-}3)$$

This formulation is the Riemann approximating summation for large N for

$$S_I(u,v) = c \int_{-I}^{I}\int_{-I}^{I} \left(1-\frac{|z_1|}{I}\right)\left(1-\frac{|z_2|}{I}\right)\rho(z_1,z_2)\cos 2\pi(uz_1+vz_2)\,dz_1\,dz_2 \qquad (3\text{-}4)$$

where c is a normalization constant and the continuous variables z_1 and z_2 have replaced the sampled variables $\frac{\tau I}{N}$ and $\frac{TI}{N}$, respectively. From Bochner's theorem it is known that

$$\rho(z_1, z_2) = R \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} D(u,v) \exp\left\{-2\pi i(uz_1 + vz_2)\right\} du\,dv \qquad (3-5)$$

where R is a constant chosen so that D(u, v) has the form of a probability density function [34, pg. 207]. Substitution of the covariance function into equation (3-4) then yields

$$S_I(u,v) = cR \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} D(u-u', v-v')\, K_I(u', v')\, du'\, dv' \qquad (3-6)$$

where K_I is the two dimensional product Fejer kernel. It is known that $S_I(u,v)$ approaches cR D(u, v) uniformly on compact sets as I approaches infinity [35, pg. 2]. Consequently, it is reasonable to assume that the variance, $\sigma^2_{F_{N,I}}(u,v)$ behaves approximately as the power spectral density, D(u, v), of the process, $\emptyset$.

The results of this analysis have been obtained by first letting the sampling interval approach zero and then letting the window of observation grow. It is important to mention that if the relaxation or correlation radius of the covariance function, ρ, is small compared to the interval of observation, then it is reasonable to assume that the variance, $\sigma^2_{F_{N,I}}(u,v)$, is already close to the power spectral density without increasing the observation window. A similar result can be obtained for the discrete two dimensional Fourier transform, F(u, v), by scaling the window of observation to unity and noting that f(x, y) is the sampled version of the continuous process, $\emptyset$. In this case

$$\sigma^2_{F_N}(u,v) = \sum_{\tau,T=0}^{N-1\,N-1} \epsilon_\tau \epsilon_T \left(1-\frac{\tau}{N}\right)\left(1-\frac{T}{N}\right)\rho\left(\frac{\tau}{N},\frac{T}{N}\right)\cos\frac{2\pi}{N}(u\tau+vT) \tag{3-7}$$

The above stochastic model indicates that for an uncorrelated process, the spectrum tends to be flat, and the variance of the spectral components of the Fourier transform of f(x, y) are fairly constant over a large range of frequencies. Conversely, if f(x, y) is a highly correlated process, the variance of F(u, v) tends to be large toward the low frequencies and falls off rapidly toward the higher frequencies. It will be assumed that the samples, f(x, y), are identically distributed with variance V^2.

It is convenient to express equation (3-7) in an expanded form in order to investigate certain limiting conditions. Therefore,

$$\begin{aligned}\rho^2_{F_N}(u,v) = \rho(0,0) &+ 2\sum_{\tau=1}^{N-1}\left(1-\frac{\tau}{N}\right)\rho\left(\frac{\tau}{N},0\right)\cos\frac{2\pi u\tau}{N} \\ &+ 2\sum_{T=1}^{N-1}\left(1-\frac{T}{N}\right)\rho\left(0,\frac{T}{N}\right)\cos\frac{2\pi vT}{N} \\ &+ 4\sum_{\tau,T=1}^{N-1\,N-1}\left(1-\frac{\tau}{N}\right)\left(1-\frac{T}{N}\right)\rho\left(\frac{\tau}{N},\frac{T}{N}\right)\cos\frac{2\pi}{N}(u\tau+vT)\end{aligned} \tag{3-8}$$

For a random process, f(x, y), which is constantly correlated in one direction, x, with correlation K, and totally uncorrelated in the other direction, the variance becomes

$$\sigma^2_{F_N}(u,v) = \rho(0,0) - K + N\,K\,\delta(u) \tag{3-9}$$

and for the case where $\rho(0,0) = K = V^2$, the variance of the identically distributed samples, $f(x,y)$, then

$$\sigma^2_{F_N}(u,v) = V^2 N\, \delta(u) \tag{3-10}$$

Equation (3-9) indicates that for highly correlated processes in one dimension the off axis variances are reduced by an amount equal to the one dimensional correlation, K, and the variances on the axis corresponding to the correlated direction are increased by an amount proportional to the correlation K. For the case where the one dimensional correlation equals the variance of the process, equation (3-10), all off axis variances are zero and large variances are experienced on the correlation axis. For constant correlation, K, in both directions the variance behaves as

$$\sigma^2_{F_N}(u,v) = \rho(0,0) - K + N^2 K\, \delta(u,v) \tag{3-11}$$

and when the correlation equals the variance of the $f(x,y)$ process, the resulting frequency sample variance is

$$\sigma^2_{F_N}(u,v) = N^2 V^2\, \delta(u,v) \tag{3-12}$$

These results indicate that a process $f(x,y)$ with constant correlation equal to its variance in all directions is a deterministic constant with a Fourier transform equal to the Kronecker delta function at the origin.

Another limiting condition that is of interest is the case of total

statistical independence of all samples in the process f(x, y). In this case the variance of F(u, v) becomes

$$\sigma^2_{F_N}(u,v) = V^2 \qquad (3\text{-}13)$$

This result indicates that for a statistically independent process all frequencies have identical variances. Under the condition of statistical independence of the samples, f(x, y), the variance is sufficient to determine the distribution of frequency components. The Central Limit Theorem applies assuming the f(x, y) samples are bounded and identically distributed, and in the limit the distribution of the function F(u, v) becomes normal [36, pg. 294].

It is of interest to determine how closely to the normal the distributions of frequency samples behave for correlation in the process f(x, y). Work has been done in this area in the one dimensional case from the point of view of a strong mixing criterion for an ergodic process [37, pg. 191]. Also, Diananda and others have proven theorems for limiting normal distributions for the r-dependent one dimensional random process [38]. Expansion to the two dimensional case is probable, but is not undertaken here.

3.2 Energy Distribution

If an original image f(x, y) ranges in magnitude in units steps from 0 to A then the maximum magnitude of a transform sample will be AN and the minimum non-zero magnitude will be 1/N. For example, the Hadamard

transform of an image for which $f(x, y) = A$ is $F(u, v) = NA\,\delta(u, v)$. The Hadamard transform of $z(x, y) = 1 \cdot \delta(x, y)$ is $F(u, v) = \frac{1}{N}$. Hence the dynamic range of transform samples in integer arithmetic is 1 to N^2A.

While the dynamic range of variables in the transform domain is extremely large, it is interesting to note that only a few points can actually take on large values since the image energy in the spatial and transform domains is identical. This energy equivalence relationship can be derived from a generalization of Parseval's relationship [39] as follows. Let

$$F(u, v)F^*(u, v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x, y)\, a(x, y, u, v) \sum_{\alpha=0}^{N-1}\sum_{\beta=0}^{N-1} f(\alpha, \beta)\, a^*(\alpha, \beta, u, v) \tag{3-14}$$

Expanding the product of the series yields

$$F(u, v)F^*(u, v) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} [f(x, y)]^2\, a(x, y, u, v)\, a^*(x, y, u, v)$$

$$+ \underbrace{\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\sum_{\alpha=0}^{N-1}\sum_{\beta=0}^{N-1}}_{\substack{x \neq \alpha \\ y \neq \beta}} f(x, y)\, f(\alpha, \beta)\, a(x, y, u, v)\, a^*(\alpha, \beta, u, v) \tag{3-15}$$

Now summing both sides over u and v gives

$$\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} |F(u, v)|^2 = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} [f(x, y)]^2 \tag{3-16}$$

as a result of the orthogonality property of the transform kernels as stated in equation (2-9). Hence there is an energy equivalence between the spatial and transform domains.

3.3 Computational Accuracy

A topic of concern for digital implementation of the transform coding system is the computational accuracy, and hence arithmetic register size, required. From considerations of the dynamic range of the transform domain it is known that the range of transform samples, other than the zero spatial frequency component, is $\pm N^2A/2$ where N is the number of elements per line and A is maximum amplitude of an element. The smallest non-zero transform sample amplitude is unity. If the register length is K bits including sign, then to prevent overflow

$$K_{MIN} = \log_2(N^2A) \tag{3-17}$$

The minimum register length for a 64 grey scale image (A = 64) is listed below as a function of N.

K_{MIN}	N
22	256
24	512
26	1024

If a smaller register size is employed than K_{MIN} overflow may occur. The probability of overflow, $\underline{P}$ is given by

$$\underline{P} = \int_{2^{K-1}}^{\frac{N^2A}{2}} P\{F(u,v)\}\,dF \tag{3-18}$$

where $P\{F(u,v)\}$ is the probability density of the amplitude of transform samples. If overflow is possible, it will generally occur at the lower

spatial frequencies. Since the expected magnitude of low spatial frequency samples is relatively large, it is often possible to normalize these samples and avoid overflow. Of course, this analysis is valid only for integer arithmetic transforms which is most likely for special purpose computer implementation. Floating point operations during image coding simulations on a general purpose digital computer alleviate the need for register length considerations.

4. Quantization of Image Transforms

To analyze the theoretical efficiency of coding the transform of a scene rather than the scene itself, it is necessary to compare the entropy of the spatial and transform domains. Andrews has shown that the entropy of a scene and its Fourier transform are identical (40). The result holds true for any transform whose Jacobian is unity. This property of image transforms, though interesting, only establishes that under ideal coding the scene and its transform can be transmitted with the same channel capacity. It remains necessary to determine quantization and coding rules for practical channels.

4.1 Quantization Methods

The selection of quantization levels can be made on the basis of minimizing the quantization error or achieving a uniform entropy for quantized sample amplitudes. In either case it is necessary to know the range and statistical distribution of the transform component to be quantized. Since this information is not available unless the transform is specified, quantization methods can only be investigated for particular transforms. Quantization methods for the Fourier and Hadamard transforms are considered in the following discussion.

Fourier transform samples are complex numbers which may be represented in real and imaginary, or magnitude and phase, form. In either case there are two components per transform sample that must be quantized. As a consequence of the statistical analysis of transform

samples of Section 3, the real, $F_R(u,v)$, and imaginary, $F_I(u,v)$, components of the Fourier transform samples are assumed to follow the same Gaussian distribution whose variance, $\sigma^2(u,v)$, is proportional to the power spectral density of the original image. Hence

$$p\{F_R(u,v)\} = \left[2\pi\,\sigma^2(u,v)\right]^{-\frac{1}{2}} \exp\left\{-\frac{F_R^2(u,v)}{2\,\sigma^2(u,v)}\right\} \tag{4-1a}$$

$$p\{F_I(u,v)\} = \left[2\pi\,\sigma^2(u,v)\right]^{-\frac{1}{2}} \exp\left\{-\frac{F_I^2(u,v)}{2\sigma^2(u,v)}\right\} \tag{4-1b}$$

If the real and imaginary components are Gaussian, the magnitude of the Fourier transform sample, $F_M(u,v)$, is Rayleigh distributed

$$p\{F_M(u,v)\} = \frac{F_M(u,v)}{\sigma^2(u,v)} \exp\left\{-\frac{F_M^2(u,v)}{2\sigma^2(u,v)}\right\} \qquad F_M(u,v) \geq 0 \tag{4-2a}$$

and its phase, $F_p(u,v)$, is uniformly distributed

$$p\{F_p(u,v)\} = \frac{1}{2\pi} \qquad -\pi \leq F_p \leq +\pi \tag{4-2b}$$

Hadamard transform samples are real, bipolar numbers which can be represented by a single component per sample. For this analysis the statistical distribution of Hadamard sample components, $F_H(u,v)$, will be considered to follow a Gaussian distribution of the form,

$$p\{F_H(u,v)\} = \left[2\pi\,\sigma^2(u,v)\right]^{-\frac{1}{2}} \exp\left\{-\frac{F_H^2(u,v)}{2\sigma^2(u,v)}\right\} \tag{4-3}$$

When the variance function, $\sigma^2(u, v)$, is not known for a particular image, or class of images, to be transformed, the function can usually be modeled without seriously affecting the quantization process. From examination of the Fourier and Hadamard transforms of a typical image, it can be deduced that the variance function should be a maximum at the origin in the transform domain, be circularly symmetric, and decrease in magnitude monotonically toward the higher spatial frequencies. A two dimensional function processing these characteristics is the Gaussian curve described by

$$\sigma^2(u,v) = S \exp\left\{ - \frac{u^2 + v^2}{p} \right\} \tag{4-4}$$

where S is an amplitude scaling constant and p is a spread control constant.

In the quantization analysis the transform sample component to be quantized (amplitude, real part, imaginary part, magnitude, or phase) is represented by the function $F_c(u, v)$. The range of the component is broken up into K positive and K negative bands separated by quantization levels Q_j ($j = 0, \pm 1, \pm 2, \cdots, \pm K$) where

$$Q_0 = 0 \tag{4-5a}$$

$$Q_K = \frac{NA}{2} \tag{4-5b}$$

$$Q_{-K} = - \frac{NA}{2} \tag{4-5c}$$

The magnitude of a sample need only be quantized over the positive scale. If a transform component falls in a band bounded by quantization levels Q_{j-1} and Q_j, the component is quantized, and subsequently reconstructed, to

to the value F_j which lies within the band. The relationship between quantization levels and reconstruction levels is given below.

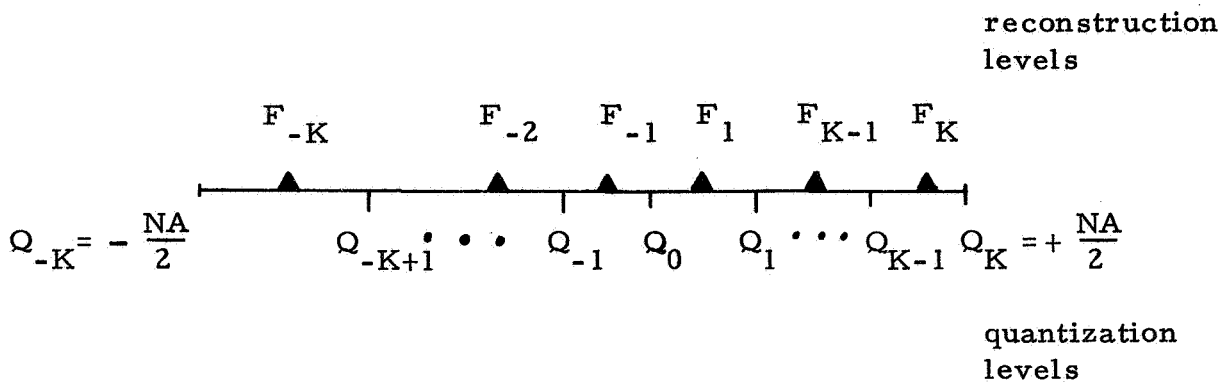


Table 4-1 lists some error criteria that might be considered in the selection of transform quantization and reconstruction levels. In general the error criteria chosen will depend upon the application of the reconstructed images; for example, whether the image is to be used for subjective viewing or photometric measurements.

For subjective viewing the relative spatial error for low brightness images provides an indication of image quality. This relative spatial error criterion is predicated upon the fact that incremental brightness changes in the reconstructed image are much more noticeable if the brightness level is low than if it is high. Thus, to minimize the relative spatial error, the density of quantization levels in the spatial domain should be greater at the lower amplitude levels. But, since the brightness of every point of a

TABLE 4-1

Quantization Error Criteria

Cumulative mean square spatial error	$\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left[f(x,y)-\tilde{f}(x,y)\right]^2$
Cumulative mean square transform error	$\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left[F(u,v)-\tilde{F}(u,v)\right]^2$
Cumulative spatial error	$\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left\|f(x,y)-\tilde{f}(x,y)\right\|$
Cumulative transform error	$\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left\|F(u,v)-\tilde{F}(u,v)\right\|$
Relative spatial error	$\frac{\left\|f(x,y)-\tilde{f}(x,y)\right\|}{\left\|f(x,y)\right\|}$
Relative transform error	$\frac{\left\|F(u,v)-\tilde{F}(u,v)\right\|}{\left\|F(u,v)\right\|}$

$\tilde{F}(u,v)$ = quantized value of $F(u,v)$

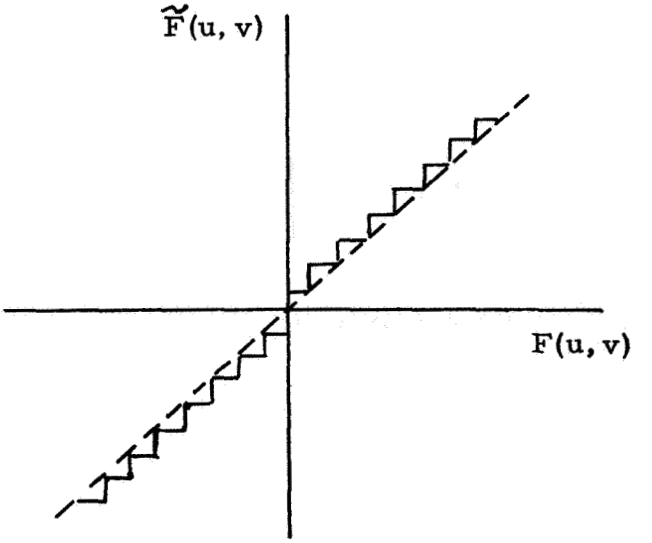
$\tilde{f}(x,y)$ = inverse transform of quantized value of $F(u,v)$

reconstructed image is a function of the amplitude of a single transform sample, then by the same reasoning, the density of quantization levels should be greater for low level transform samples. From psychological tests, it is known that the human viewer is very sensitive to the location of high frequency brightness transitions, but relatively insensitive to their actual magnitude. In fact images which have been "crispened" by high pass filtering often appear preferable to the original image. From this characteristic of subjective viewing it would seem that the density of quantization levels at low transform sample amplitudes should be greater at the higher spatial frequencies than at the lower spatial frequencies. Thus, from the standpoint of subjective quality, the "best" quantizer should have a nonlinear characteristic such that the density of quantization levels over the range of the amplitude of the transform sample component to be quantized is:

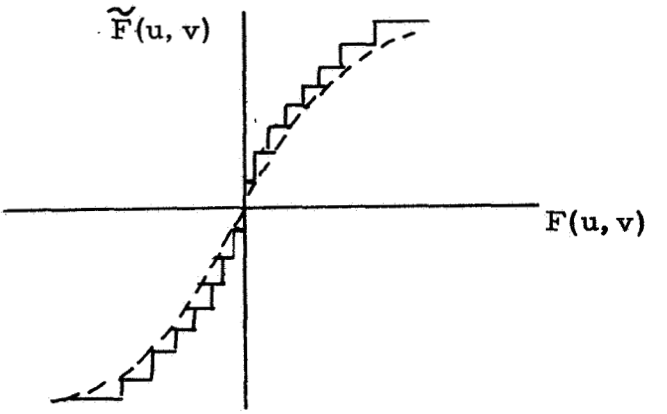
a) greater at its lower values for a given spatial frequency

b) greater at the higher spatial frequencies for a given amplitude

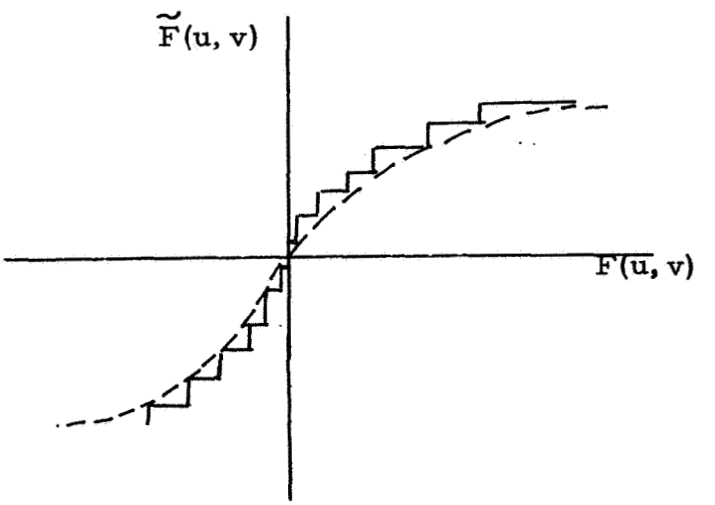
Figure 4-1 exhibits several quantization laws that are useful for quantization under a subjective viewing error criterion. The uniform or linear quantizer is commonly employed for quantization of the phase of Fourier transform samples. The Gaussian quantizers follow the mathematical function



a) Linear Quantizer



b) Gaussian Quantizer



c) Logarithmic Quantizer

Figure 4-1. Qunatization Rules

$$\tilde{F}_c(u,v) = \operatorname{erf}\left[\frac{F(u,v)}{\sqrt{2}\,K(u,v)}\right] \tag{4-6}$$

between the transform component, $F_c(u,v)$, and the quantized transform component, $F_c(u,v)$, where

$$\operatorname{erf}(x) \equiv \frac{2}{\sqrt{\pi}} \int_0^x \exp\{-z^2\}\, dz \tag{4-7}$$

is the Gaussian error function and $K(u,v)$ is a two dimensional positive function monotonically decreasing with u and v. The Gaussian quantizer has the desired property that the spacing of quantization levels is closer for the lower amplitudes of $F_c(u,v)$ at a given spatial frequency u, v and closer for the higher values of v and u at a given value of $F_c(u,v)$. Furthermore, if $K(u,v)$ is set equal to the standard deviation of the transform samples, $\sigma(u,v)$, then the probability that a transform sample will be quantized to a given reconstruction level will be the same for all quantization levels. This results in a uniform entropy for all reconstruction levels, and therefore, a constant word length code may be used for each quantized sample. The logarithmic quantizer follows the function

$$\tilde{F}_c(u,v) = \ell n\,[\,K(u,v)(F_c(u,v)+1)] \tag{4-8}$$

in the positive quadrant and the inverted and reversed version of the function in the negative quadrant. This function has the same general characteristics as the Gaussian quantizer, but does not produce an equal entropy for quantized samples.

If photometric measurements are to be made on an image the cumulative mean square spatial error is a common fidelity criterion. For such a situation the quantization levels in the transform domain must be selected to minimize the cumulative mean square error in the spatial domain. Let

$$\mathcal{E}_S \equiv \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \overline{[f(x,y) - \tilde{f}(x,y)]^2} \tag{4-9}$$

represent the cumulative mean square spatial error where $\tilde{f}(x,y)$ is the image reconstruction from the quantized transform samples, $\tilde{F}(u,v)$. Then, in a matrix formulation

$$\mathcal{E}_S = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \overline{\{[B][F][B] - [B][\tilde{F}][B]\}^2} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \overline{[B][F-\tilde{F}]^2[B]} \tag{4-10}$$

where [B] represents the reverse transformation matrix.

Minimization of $\mathcal{E}_S$ in the spatial domain therefore can be accomplished by the minimization of the mean square error, $\mathcal{E}(u,v) \equiv [F-\tilde{F}]^2$, in the transform domain for all spatial frequencies.

In the case of the Fourier transform the mean square error of each component of a transform sample must be minimized. The mean square error of a transform component may be written in explicit form as

$$\mathcal{E}(u,v) = \mathcal{E}_+(u,v) + \mathcal{E}_-(u,v) \tag{4-11}$$

where

$$\mathcal{E}_{+}(u,v) = \sum_{j=1}^{K} \int_{Q_{j-1}}^{Q_j} (F_c - F_j)^2 \, p(F_c) \, dF_c$$

and

$$\mathcal{E}_{-}(u,v) = \sum_{j=-1}^{-K} \int_{Q_{j+1}}^{Q_j} (F_c - F_j)^2 \, p(F_c) \, dF_c$$

where $p(F_c)$ is the probability density of the transform sample component to be quantized. If $p(F_c)$ is a symmetrical probability density about $Q_0 = 0$, then $\mathcal{E}_{+}(u,v)$ equals $\mathcal{E}_{-}(u,v)$. Regardless of the form of $p(F_c)$ the quantization rule determined by the minimization of $\mathcal{E}_{+}(u,v)$ is the same as that determined from $\mathcal{E}_{-}(u,v)$ because of the symmetry of the quantization scale. Hence, only $\mathcal{E}_{+}(u,v)$ will be considered in the following analysis.

For a large number of quantization levels the probability density of the transform samples may be represented by a constant value, $p(F_j)$, over the quantization band. Hence,

$$\mathcal{E}_{+}(u,v) \approx \sum_{j=1}^{K} p(F_j) \int_{Q_{j-1}}^{Q_j} (F_c - F_j)^2 \, dF_c = \frac{1}{3} \sum_{j=1}^{K} p(F_j)[(Q_j - F_j)^3 - (Q_{j-1} - F_j)^3] \tag{4-12}$$

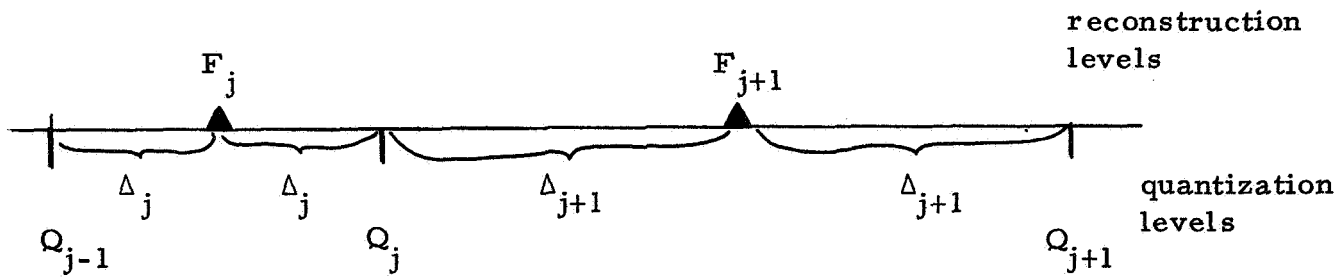
The optimum placing of the reconstruction level F_j within the range Q_{j-1} to Q_j can be determined by minimization of $\mathcal{E}_{+}(u,v)$ with respect to F_j. Setting

$$\frac{d\mathcal{E}_{+}(u,v)}{dF_j} = \frac{1}{3} \, p(F_j) \{ -3(Q_j - F_j)^2 + 3(Q_{j-1} - F_j)^2 \} = 0 \tag{4-13}$$

yields

$$F_j = \frac{Q_j + Q_{j-1}}{2} \tag{4-14}$$

Therefore the condition for minimizing $\mathcal{E}_+(u, v)$ in the range Q_{j-1} to Q_j is to place the reconstruction level F_j at the midpoint between each pair of quantization levels. The general relationship between reconstruction and quantization levels for the constraint of equation (4-14) is shown below.



Hence,

$$Q_j = F_j + \Delta_j \tag{4-15}$$

and

$$Q_{j-1} = F_j - \Delta_j \tag{4-16}$$

where

$$\Delta_j \equiv \frac{Q_j - Q_{j-1}}{2} \tag{4-17}$$

With the value for F_j from equation (4-14) substituted into equation (4-12) the mean square transform error becomes

$$\mathcal{E}_{+}(u,v) \approx \frac{1}{12} \sum_{j=1}^{K} p[F_j] [Q_j - Q_{j-1}]^3 \tag{4-18}$$

Now, by the definition of the integral of a function it is possible to write

$$\sum_{j=1}^{K} \left\{p^{1/3}[F_j]\right\}[Q_j - Q_{j-1}] = \int_{0}^{+\frac{NA}{2}} p^{\frac{1}{3}}[F_c]\, dF_c = \mathcal{K} \tag{4-19}$$

where the value of the integral, $\mathcal{K}$, is a constant only depending upon its limits. Thus, the problem of minimizing $\mathcal{E}_{+}(u,v)$ with respect to the reconstruction levels Q_j reduces to minimizing the sum of cubes of a variable in equation (4-18) subject to the constraint of equation (4-19) that the sum of the variables is constant. By the method of Lagrange multipliers $\mathcal{E}_{+}(u,v)$ is minimized when $\left\{p^{1/3}[F_j]\right\}[Q_j - Q_{j-1}]$ is identical for all K quantization bands. Under this condition

$$\left\{p^{1/3}[F_j]\right\}[Q_j - Q_{j-1}] = \frac{\mathcal{K}}{K} \tag{4-20}$$

and

$$[\mathcal{E}_{+}(u,v)]_{MIN} = \frac{1}{12} \frac{\mathcal{K}^3}{K} = \frac{1}{12K^2} \left(\int_{0}^{+\frac{NA}{2}} \left\{p^{1/3}[F]\right\} dF \right)^3 \tag{4-21}$$

The quantization levels can be determined from the formula

$$Q_j = 2\Delta_1 + 2\Delta_2 + \cdots + 2\Delta_{j-1} + \Delta_j \tag{4-22}$$

where from equation (4-20)

$$\Delta_j = \frac{\chi}{2K\left\{p \cdot [F_j]\right\}^{\frac{1}{3}}} \tag{4-23}$$

Therefore,

$$Q_j = \frac{\chi}{K}\left[\frac{1}{\left\{p\ [F_1]\right\}^{\frac{1}{3}}} + \frac{1}{\left\{p\ [F_2]\right\}^{\frac{1}{3}}} \cdots + \frac{1}{\left\{p\ [F_{j-1}]\right\}^{\frac{1}{3}}}\right] \tag{4-24}$$

This series may be approximated by the normalized integral [47]

$$Q_j = \frac{\frac{NA}{2}\int_0^{j\frac{NA}{2K}} \left\{p\quad [F]\right\}^{-\frac{1}{3}} dF}{\int_0^{\frac{NA}{2}} \left\{p\quad [F]\right\}^{-\frac{1}{3}} dF} \tag{4-25}$$

As an example of the computation of quantization levels by equation (4-25) consider the case for which $p(F) = \frac{1}{NA}$. The quantization levels are then $Q_j = \frac{jNA}{2K}$ for $j = 1, 2, \cdots, K$.

If $p(F)$ is a Gaussian distribution with variance $\sigma^2(u, v)$ then Q_j is given by

$$Q_j = \frac{\frac{NA}{2}\int_0^{\frac{jNA}{2K}} \exp\left\{\frac{F^2}{6\sigma^2}\right\} dF}{\int_0^{\frac{NA}{2}} \exp\left\{\frac{F^2}{6\sigma^2}\right\} dF} \tag{4-26}$$

The quantization levels computed from equation (4-26) are more closely spaced for j small in the same general manner as the Gaussian or logarithmic quantizer.

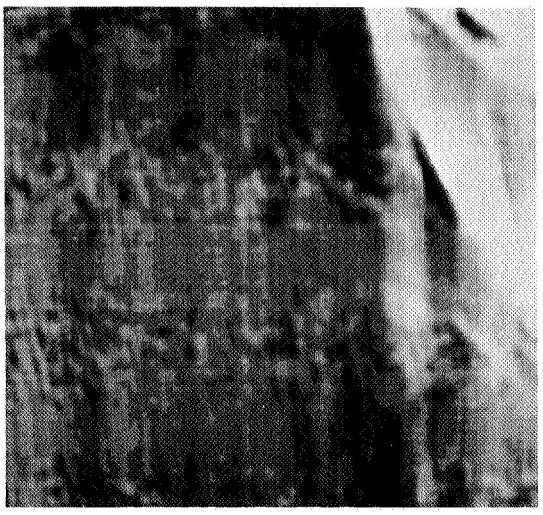
Examples of image transform quantization for the Fourier and Hadamard transform using the uniform and Gaussian quantization laws are presented in the following subsection.

4.2 Quantization Experiments

Reconstructions of the Fourier transform of the Surveyor spacecraft scenes with linear and Gaussian quantization are shown in Figure 4-2. For both quantization rules 64 levels have been employed. The Gaussian quantization rule utilized a Gaussian shaped variance parameter with a spread control constant, p = 500. Results with the linear quantizer for the footpad scene are poor because of the large quantization errors at high spatial frequencies. The Gaussian quantizer reconstructions for the footpad, boom, and box scenes show negligible image degradation.

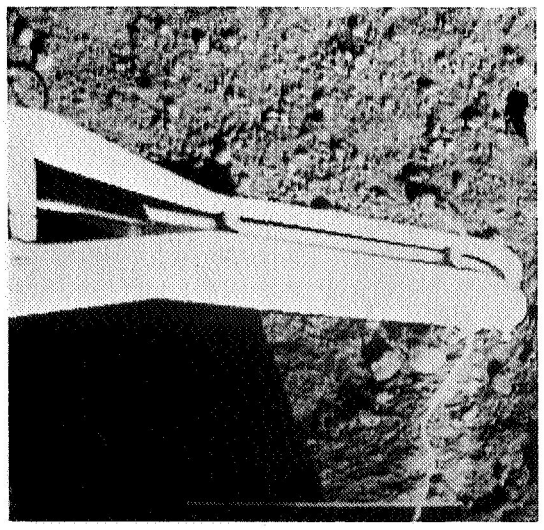
Figure 4-3 illustrates tests to determine the effect of few quantization levels with the Gaussian quantizer In these tests reconstructions have been made with the Fourier transform samples quantized to 32 and 16 levels with a Gaussian shaped variance function with p = 500. The loss of resolution in these pictures is due to the quantization errors at high spatial frequencies.

If the Gaussian quantization rule is to be practical it is imperative that a variance function can be chosen for a wide class of scenes without detailed knowledge of the content of these scenes. Figure 4-2 shows that the

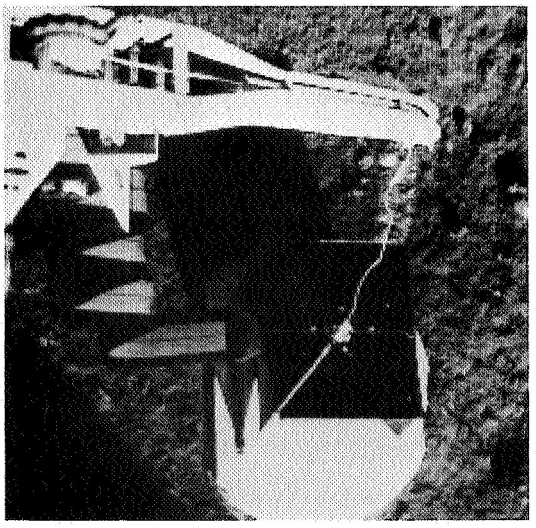


a) Inverse Fourier Transform of Linearly Quantized Fourier Transform of Footpad

b) Inverse Fourier Transfrom of Gaussianly Quantized Fourier Transform of Footpad, p = 500



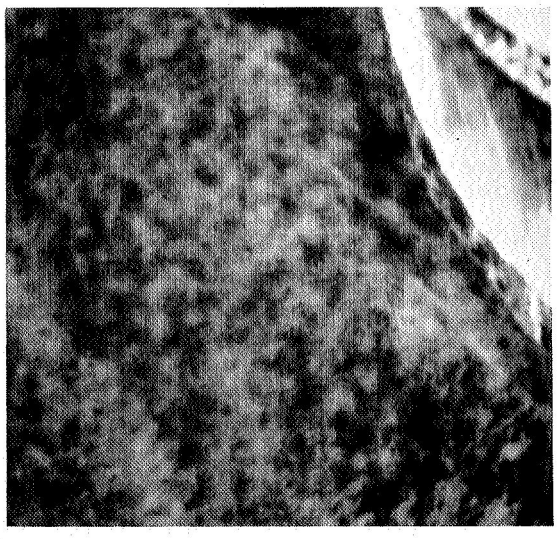
c) Inverse Fourier Transform of Gaussianly Quantized Fourier Transform of Boom, p = 500



d) Inverse Fourier Transform of Gaussianly Quantized Fourier Transform of box, p = 500

Figure 4-2. 64 Level Quantization of Fourier Transform

a) Inverse Fourier Transform of Gaussianly Quantized Fourier Transform - 32 levels



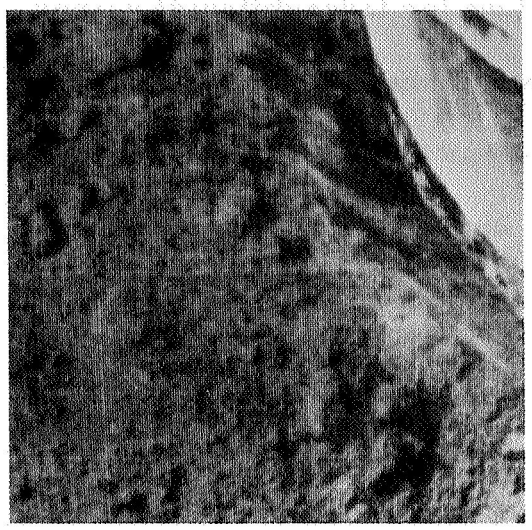
b) Inverse Fourier Transform of Gaussianly Quantized Fourier Transform - 16 levels

Figure 4-3. 32 and 16 level Gaussian Quanitzation of Fourier Transform of Footpad, p = 500.

Gaussian shaped variance function with the spread control parameter p = 500 provides satisfactory reconstructions for three different scenes. It is also of interest to determine the effect of changes in the variance function. Figure 4-4 contains reconstructions using the Gaussian quantizer with 64 quantization levels for a Gaussian shaped variance function with a spread parameter of p = 250 and 1000, and with a $|(\text{sinc } au)(\text{sinc } av)|$ shaped variance function. These experiments indicate that fortunately the performance of the Gaussian quantizer is relatively insensitive to the exact mathematical form of the variance function.

The same set of quantization experiments has been performed for quantization of Hadamard transform samples with essentially the same results and conclusions. The results of these experiments shown in Figures 4-5 to 4-7 are self explanatory.

The conclusion of the quantization experiments for the Fourier and Hadamard transform is that good quality reconstructions are possible when the transform samples have been quantized to as few as 64 levels using the Gaussian quantization rule and a Gaussian shaped variance function with an appropriate spread control parameter.

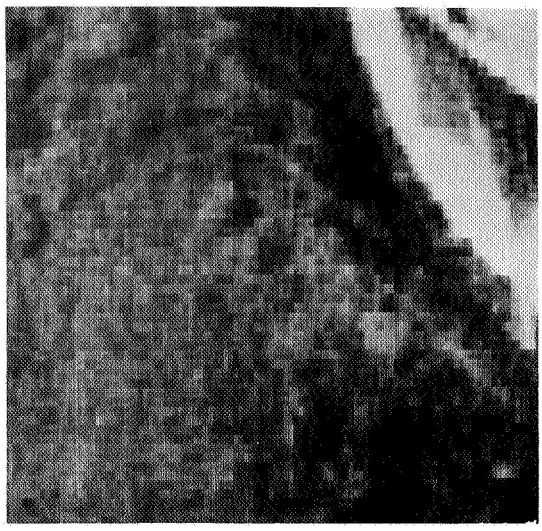


a) Inverse Fourier Transform of Gaussianly Quantized Fourier Transform with a Narrow Gaussian Shaped Variance Function, p = 250

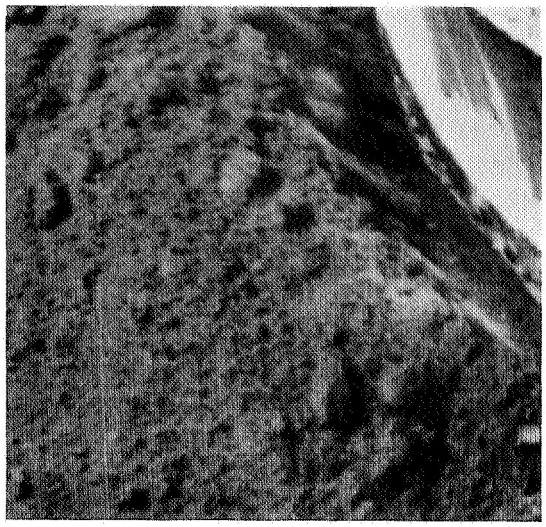
b) Inverse Fourier Transform of Gaussianly Quantized Fourier Transform with a Wide Gaussian Shaped Variance Function, p = 1000

c) Inverse Fourier Transform of Gaussianly Quantized Fourier Transform with a $|(\text{sinc } au)(\text{sinc } av)|$ Shaped Variance Function

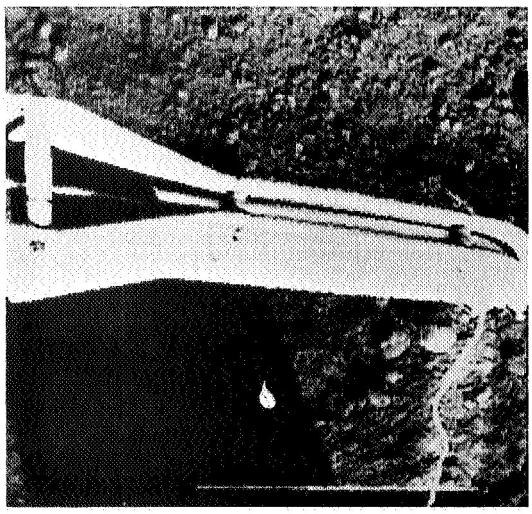
Figure 4-4. 64 Level Gaussian Quantization of Fourier Transform of Footpad with Different Variance Functions



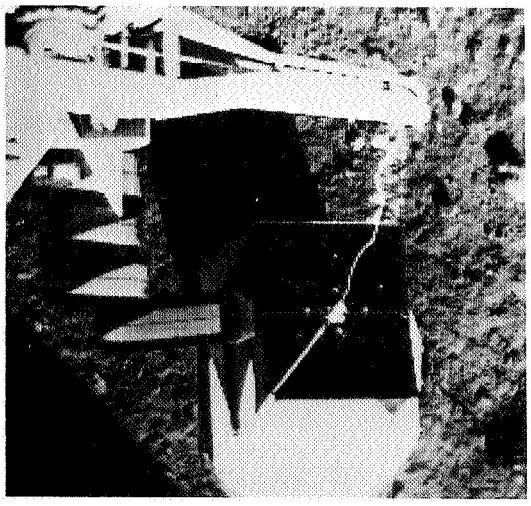
a) Hadamard Transform of Linearly Quantized Hadamard Transform of Footpad



b) Hadamard Transform of Gaussianly Quantized Hadamard Transform of Footpad, p = 1500

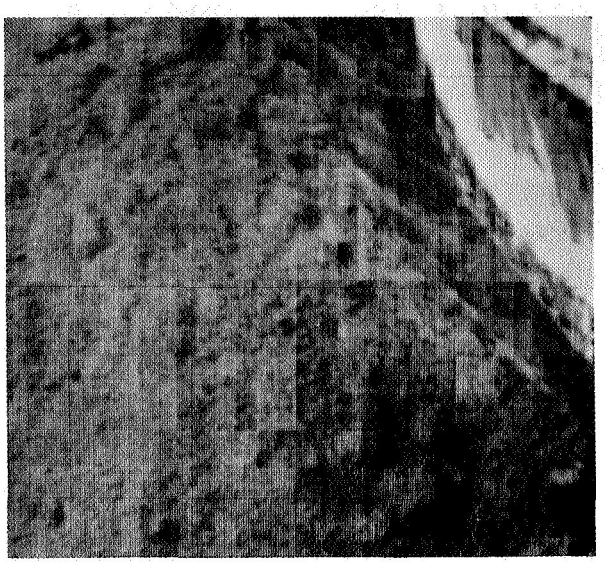


c) Hadamard Transform of Gaussianly Quantized Hadamard Transform of Boom, p = 1500



d) Hadamard Transform of Gaussianly Quantized Hadamard Transform of Box, p = 1500

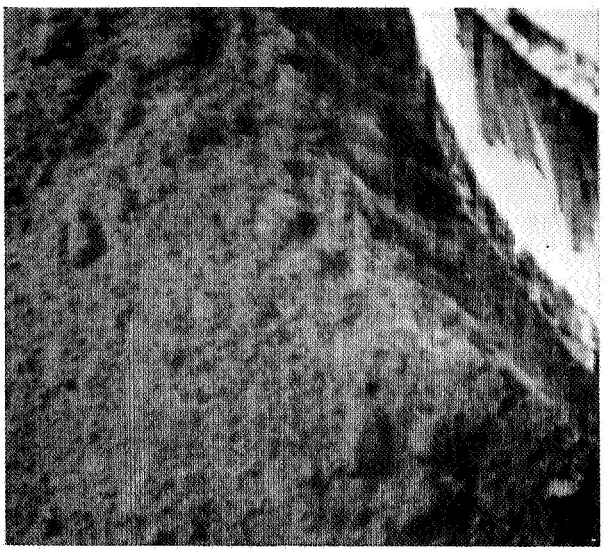
Figure 4-5. 64 Level Quantization of Hadamard Transform



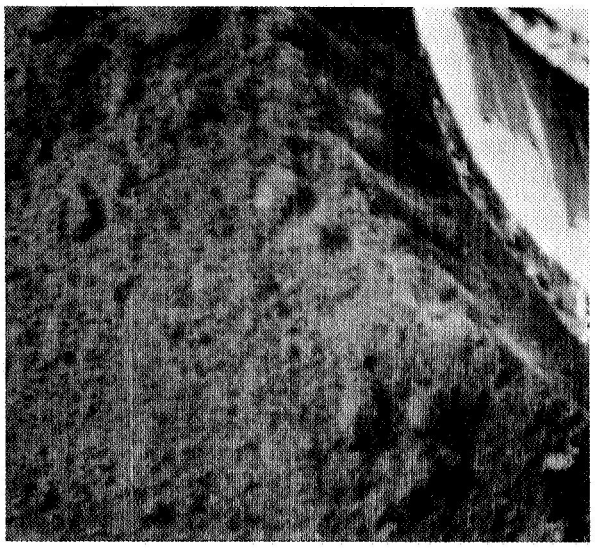
a) Hadamard Transform of Gaussianly Quantized Hadamard Transform - 32 levels

b) Hadamard Transform of Gaussianly Quantized Hadamard Transform - 16 levels

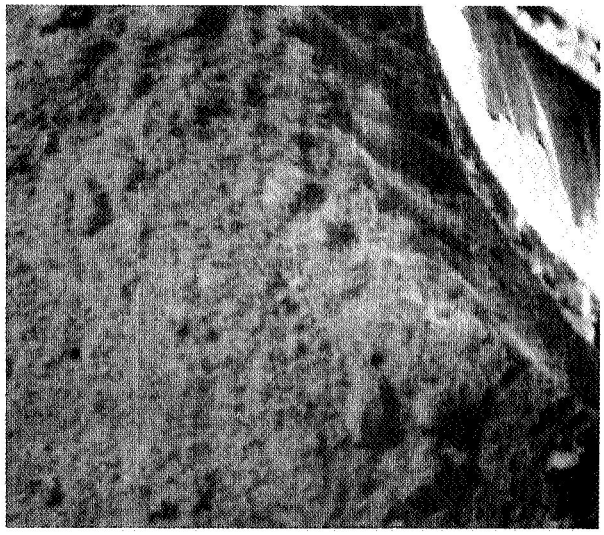
Figure 4-6. 32 and 16 Level Gaussian Quantization of Hadamard Transform of footpad, p = 1500



a) Hadamard Transform of Gaussianly Quantized Hadamard Transform with p = 500



b) Hadamard Transform of Gaussianly Quantized Hadamard Transform with p = 1000



c) Hadamard Transform of Gaussianly Quantized Hadamard Transform with p = 2000

d) Hadamard Transform of Gaussianly Quantized Hadamard Transform with p = 5000

Figure 4-7. 64 Level Gaussian Quantization of Hadamard Transform of Footpad with different Gaussian shaped Variance Functions

5. Bandwidth Reduction

Transmission of the transform of an image rather than the image itself opens up a wide area of investigation for the development of image transform bandwidth reduction techniques. Such techniques may be divided into two categories: those which are based upon the unique structure of the energy distribution in the transform plane, and those that seek to apply conventional spatial domain bandwidth reduction methods to the transform domain in a manner rather independent of its energy distribution. In general, the former class of methods provide the best performance. Attempts to apply spatial domain bandwidth reduction techniques to transform samples have not prove successful [9] because of the large dynamic ranges of transform samples and their relative lack of correlation with one another.

5.1 Transform Sampling

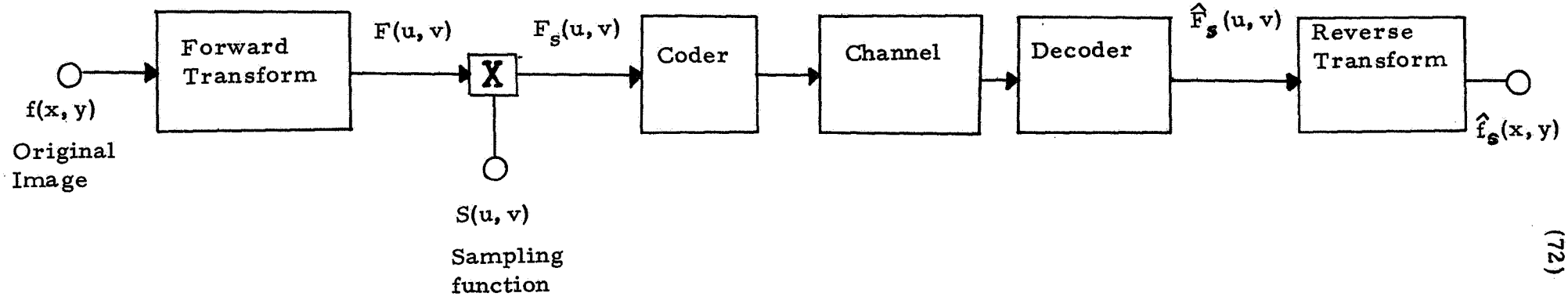
Many transform bandwidth reduction techniques can be analyzed from the viewpoint of two dimensional sampling. Figure 5-1 illustrates a generalized block diagram of a transform sampling system. The forward transform of an image, $F(u, v)$, is multiplied by a two dimensional sampling function, $S(u, v)$, which takes on the values zero or one according to some apriori or adaptive rule. Several transform sampling methods are listed in Table 5-1. These methods will be considered individually in subsequent subsections.

With reference to figure 5-1, the sampled transform, $F_s(u, v)$,

Table 5-1

Classifications of Transform Sampling Methods

Description	Sampling Function S(u, v)	Conditions
checkerboard sampling	$\frac{1 + (-1)^{u+v}}{2}$	odd samples set to zero
random sampling	1	with probability p
	0	with probability 1-p
zonal sampling	1	u, v ϵ sampling region
	0	u, v $\notin$ sampling region
threshold sampling	1	if $\lvert F(u,v)\rvert > M_T(u,v)$
	0	if $\lvert F(u,v)\rvert \leq M_T(u,v)$



(72)

Figure 5-1. Transform Domain Sampling

is simply

$$F_s(u,v) = F(u,v)\, S(u,v) \tag{5-1}$$

After decoding $\hat{F}_s(u,v)$ is reconstructed and the reverse transform produces $\hat{f}_s(x,y)$. The effect of channel errors on reduced bandwidth signals is an important topic which will be considered in the next section. For this analysis the transmission will be assumed to be errorless. With this assumption the factor of greatest importances becomes the closeness with which the reverse transform of $F_s(u,v)$ approximates the original image, $f_s(x,y)$. Taking the reverse transform of $F_s(u,v)$ yields

$$f_s(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v)\, S(u,v)\, b(x,y,u,v) \tag{5-2}$$

Since

$$F(u,v) = \sum_{\alpha=0}^{N-1} \sum_{\beta=0}^{N-1} f(\alpha,\beta)\, a(\alpha,\beta,u,v) \tag{5-3}$$

the system output image may be expressed as

$$f_s(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} S(u,v)\, b(x,y,u,v) \sum_{\alpha=0}^{N-1} \sum_{\beta=0}^{N-1} f(\alpha,\beta)\, a(\alpha,\beta,u,v) \tag{5-4}$$

Upon changing the order of summation

$$f_s(x,y) = \sum_{\alpha=0}^{N-1} \sum_{\beta=0}^{N-1} f(\alpha,\beta) \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} S(u,v)\, a(\alpha,\beta,u,v)\, b(x,y,u,v) \tag{5-5}$$

From equation (2-8) the product of the transformation kernels is

$$a(\alpha, \beta, u, v)\, b(x, y, u, v) = \frac{1}{N}\, b(x-\alpha, y-\beta, u, v) \tag{5-6}$$

The second summation is then recognized to be the reverse transform of $s(u, v)$ evaluated at the point $\alpha - x, \beta - y$ in the spatial domain. Hence,

$$f_s(x, y) = \frac{1}{N} \sum_{\alpha=0}^{N-1} \sum_{\beta=0}^{N-1} f(\alpha, \beta)\, s(x-\alpha, y-\beta) \equiv f(x, y) \circledast s(x, y) \tag{5-7}$$

represents the spatial convolution denoted by the symbol $\circledast$, of the original image, $f(x, y)$, with the reverse transform of the sampling function, $S(u, v)$.

In general, any sampling function can be expressed as

$$S(u, v) = \frac{1 + R(u, v)}{2} \tag{5-8}$$

where $R(u, v)$ takes on the value ± 1 as a function of the spatial frequencies u and v. The reverse transform of the sampled transform domain is then

$$f_s(x, y) = \frac{1}{2} [f(x, y) + f(x, y) \circledast r(x, y)] \tag{5-9}$$

where $r(x, y)$ is the reverse transform of $R(u, v)$. Thus, the reconstruction of the sampled image is composed of the original image plus some additive interference that is dependent upon the form of the original image and the sampling function.

As an example of deterministic sampling, consider a sampling function

$$S(u, v) = \frac{1 + (-1)^{u+v}}{2} \tag{5-10}$$

which samples the Fourier transform of an image in a checkerboard

pattern. For this case

$$R(u,v) = (-1)^{u+v} = \exp\{i\pi(u+v)\} \tag{5-11}$$

and its inverse Fourier transform is

$$r(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \exp\{i\pi(u+v)\} \exp\left\{\frac{-2\pi i}{N}(ux+vy)\right\} \tag{5-12}$$

or

$$r(x,y) = \delta\left(x+\frac{N}{2}, y+\frac{N}{2}\right) \tag{5-13}$$

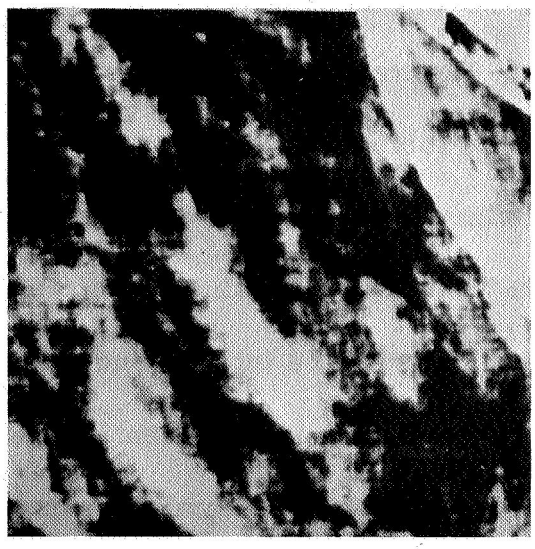
Hence, the reconstructed image

$$f_s(x,y) = \frac{1}{2}\left[f(x,y) + f\left(x+\frac{N}{2}, y+\frac{N}{2}\right)\right] \tag{5-14}$$

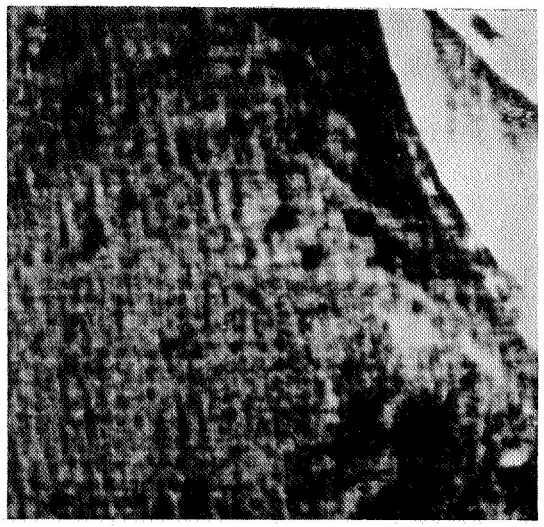
is composed of the original image overlaid by the original shifted horizontally and vertically by one-half its size. Figure 5-2a illustrates the experimental verification of this effect for the footpad scene.

A non-deterministic sampling procedure that has been considered is one in which $R(u,v)$ is a random variable assuming the values ± 1. If this random variable is highly uncorrelated, the additive interference is spread out over the reconstructed image. This technique has been investigated for random sampling of the Fourier transform of an image in which 50% of the transform components have been sampled at random positions to yield a bandwidth reduction of 2 : 1. The reconstructed image of the footpad scene is shown in figure 5-2b. With this type of random sampling the convolutional interference produces a significant amount of image

a) Checkerboard Sampling



b) Random Sampling



c) Random Sampling of High Spatial Frequencies Only

Figure 5-2. Checkerboard and Random Sampling of Fourier Transform

degradation. Distortion in this image is due principally to the convolution of the high brightness, and low spatial frequency, portion of the footpad over the image surface. To overcome this difficulty only the highest 90% of the spatial frequencies of the image were randomly sampled. The low spatial frequencies were completely sampled. The reconstruction in figure 5-2c for this type of sampling shows some improvement, but the image distortion is still severe.

5.2 Zonal Sampling

In most scenes of interest there is a fairly high degree of correlation between adjacent image elements. For these types of images the energy in the transform plane tends to be clustered at certain spatial frequencies.

Figure 5-3 illustrates the percentage of energy within a circle centered at the origin of the Fourier transform plane for the three Surveyor spacecraft scenes. For all three scenes 95% of the image energy is contained in 1% or less of the Fourier domain samples. With an image energy distribution such as that shown in figure 5-3 the most obvious method of conserving bandwidth is simply to not transmit the high spatial frequency information. Discarding the high spatial frequencies is equivalent to passing the image through a circular, zonal low pass filter; the result is a loss of focus. Figures 5-4 and 5-5 show the effect of zonal low pass sampling of the Surveyor spacecraft footpad and box scenes. These experiments support the widely known fact that the high frequency

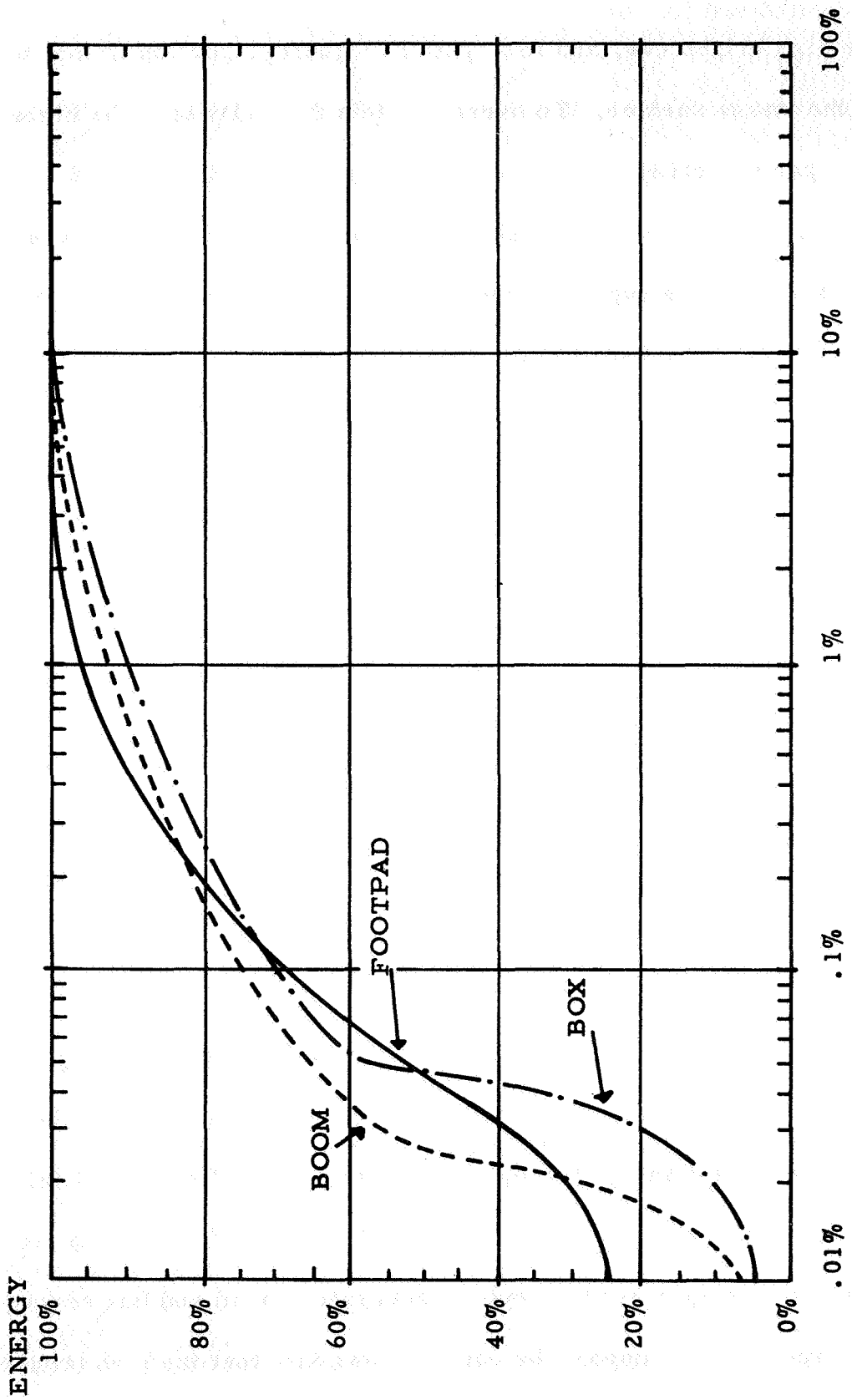


Figure 5-3. Image Energy as a Function of Percentage of Fourier Transform Samples Within a Circular Zone.

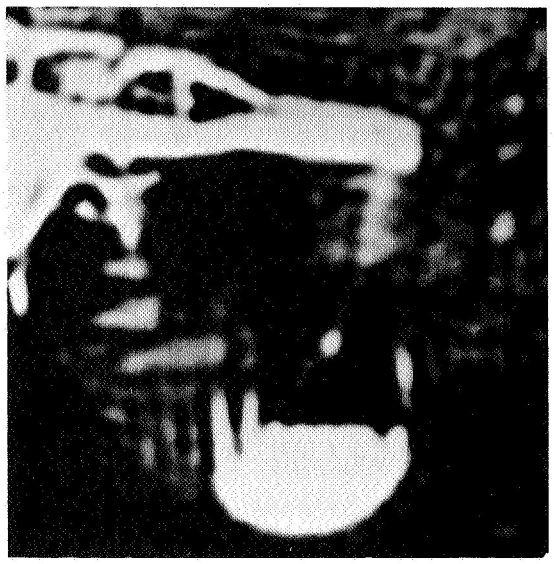
a) 99.8% energy transmitted, 32:1 BWR

b) 99.9% energy transmitted, 8:1 BWR

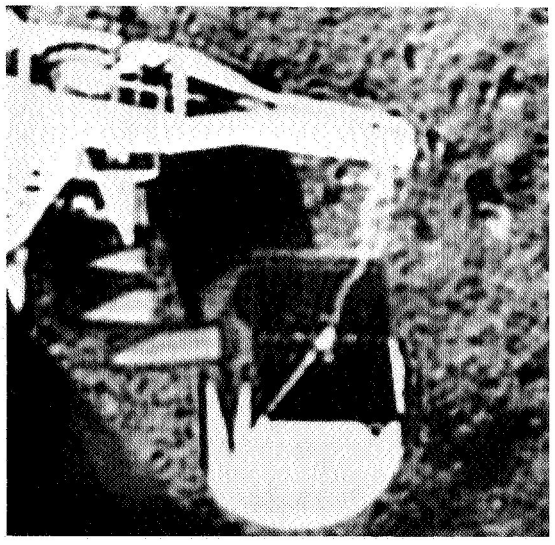
c) 99.99% energy transmitted, 4:1 BWR

Figure 5-4. Low Pass Zonal Fourier Transform Sampling - Footpad

a) 98.3% energy transmitted,
32:1 BWR



b) 99.8% energy transmitted,
8:1 BWR



c) 99.9% energy transmitted,
4:1 BWR

Figure 5-5. Low Pass Zonal Fourier Transform Sampling - Box

brightness transitions are important even though they are relatively few in number and contain a low proportion of the image energy. However, if some degree of resolution loss is acceptable, zonal low pass filtering of the Fourier domain does yield relatively large bandwidth reductions.

Zonal low pass sampling or filtering can also be performed in the Hadamard transform domain. Figure 5-6 illustrates a reconstruction of the lowest 25% of the Hadamard domain spatial frequencies of the footpad scene. The image degradation tends to be more noticeable for zonal filtering of the Hadamard transform than for the Fourier transform for the same bandwidth reduction factor because of the rectangular shape of the two dimensional Hadamard reconstruction waveforms. The eye is very sensitive to the presence of sharp brightness transitions within an image. With the Hadamard transform all transitions occur within one element, whereas in the Fourier transform the brightness transitions are spread over many elements since the reconstruction waveforms are two dimensional sinusoids.

5.3 Threshold Sampling

The difficulty with the zonal filter sampling method of bandwidth reduction is that large magnitude samples are indiscriminately discarded. An obvious answer to this problem is to code only those samples whose magnitudes are above a given threshold level. With this coding method it becomes necessary to provide information as to the location of significant samples.

Figure 5-6. Low Pass Zonal Hadamard Transform Sampling - Footpad

Selection of a threshold level for a given transform is generally a compromise between the number of samples deleted and the error resulting from the deletion of samples. If a constant threshold is chosen, then the maximum magnitude of deleted samples will be independent of spatial frequency, but the probability of deleting a given sample will usually be a function of its spatial frequency. On the other hand, if the threshold is chosen to be linearly dependent upon the variance of samples, the deletion probability in most cases will be constant for all samples, but the deletion error will be a function of spatial frequency. The "best" threshold can only be determined analytically for a given error criterion.

For transform threshold coding the "best" transform is one which maximizes the number of transform samples which are zero or near zero when the error criterion is satisfied. It has been pointed out in Section 2 that to minimize the mean square error between an original signal and a transform reconstruction for threshold sample deletion, the optimum transform is composed of Eigenvectors of the correlation matrix of the data. Since mean square error has not proven an effective measure of error between images, the usefulness of this transform remains in doubt. Further investigation of the application of the Eigenvector transform to image coding is required.

For a given transform the expected bandwidth reduction achievable with threshold sampling of the transform domain can be estimated if the probability distribution of the magnitude of transform samples is known. Let $M(u, v) \equiv |F(u, v)|$ = magnitude of a transform sample and

$P[M(u,v)]$ = probability distribution of the magnitude of transform samples, then the probability, $P(u,v)$, that the magnitude of a transform sample of spatial frequency (u,v) is greater than a threshold level. $M_T(u,v)$, is

$$P(u,v) = \int_{M_T(u,v)}^{\infty} P[M(u,v)]\,dM \tag{5-15}$$

The expected number of samples above the threshold, N_T, is then given by

$$N_T = \sum_u \sum_v P(u,v) \tag{5-16}$$

where the limits of the summation are dependent upon the type of transform employed.

Consider first Fourier threshold sampling. If it is assumed that the probability density of the real and imaginary components of transform samples are Gaussian with a variance function $\sigma^2(u,v)$, then the magnitude of a transform sample becomes Rayleigh distributed with a distribution

$$P[M(u,v)] = \frac{M}{\sigma^2(u,v)} \exp\left\{-\frac{M^2}{2\sigma^2(u,v)}\right\} \qquad M \geq 0 \tag{5-17}$$

Then the probability that the sample magnitude is greater than the threshold, $M_T(u,v)$, is

$$P(u,v) = \exp\left\{\frac{-[M_T(u,v)]^2}{2\sigma^2(u,v)}\right\} \tag{5-18}$$

As a result of the conjugate symmetry property of the Fourier transform,

only one half of the transform samples need be considered. The expected number of these samples above the threshold is given by

$$N_T = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \exp \left\{ \frac{-[M_T(u,v)]^2}{2\sigma^2(u,v)} \right\} \tag{5-19}$$

In the special case for which the threshold is linearly dependent upon the variance, i.e.,

$$M_T(u,v) = k_T \sigma(u,v) \tag{5-20}$$

where k_T is a constant, the expected number of samples above the threshold is simply

$$N_T = \frac{N^2}{2} \exp \left\{ -\frac{k_T^2}{2} \right\} \tag{5-21}$$

The transform sample reduction is then

$$\frac{N^2/2}{N_T} = \exp \left\{ \frac{k_T^2}{2} \right\} \tag{5-22}$$

For the Hadamard transform the magnitude of the samples can be modeled as a Gaussian distribution.

$$P[M(u,v)] = [2\pi\sigma^2(u,v)]^{-1/2} \exp \left\{ -\frac{M^2}{2\sigma^2(u,v)} \right\} \tag{5-23}$$

Then the probability that a transform sample exceeds the threshold becomes

$$P(u,v) = \left[1 - \text{erf}\left(\frac{M_T(u,v)}{\sqrt{2}\,\sigma(u,v)}\right)\right] \tag{5-24}$$

where

$$\text{erf}\{x\} \equiv \int_0^x e^{-y^2}\,dy \tag{5-25}$$

is the Gaussian error function. The expected number of transform samples above the threshold is

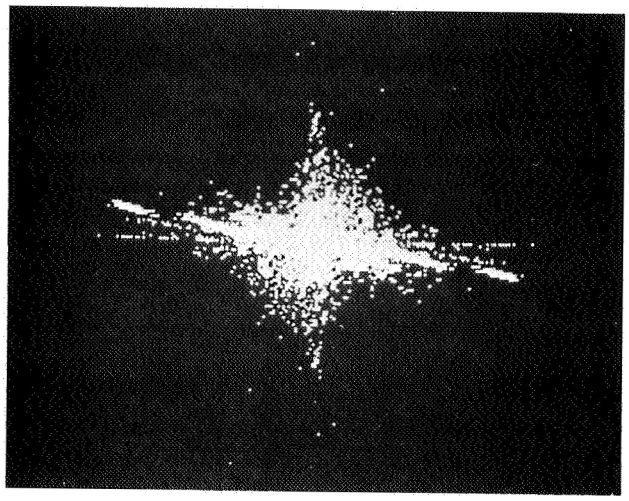
$$N_T = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left[1 - \text{erf}\left(\frac{M_T(u,v)}{\sqrt{2}\,\sigma(u,v)}\right)\right] \tag{5-26}$$

And for the special case for which $M_T(u,v) = k_T\sigma(u,v)$, the transform sample reduction becomes

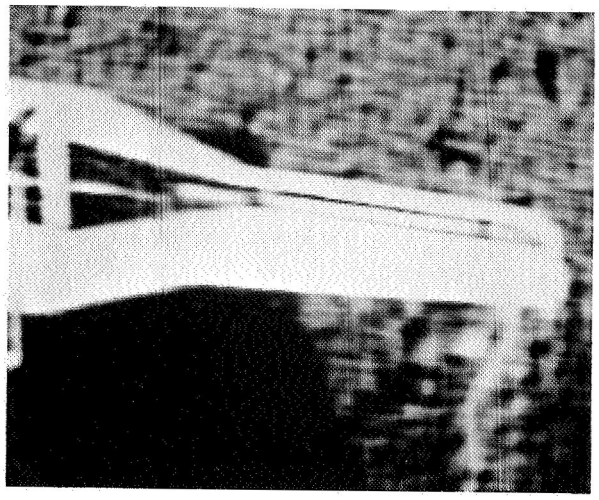
$$\frac{N^2}{N_T} = \frac{1}{\left[1 - \text{erf}\left(\frac{k_T}{\sqrt{2}}\right)\right]} \tag{5-27}$$

From the preceding analysis it is seen that if the variance factor $\sigma^2(u,v)$ is known for a particular class of images, the expected bandwidth reduction factor can be easily computed for Fourier and Hadamard threshold sampling with a given threshold level. The amount of image degradation for a given threshold level must be determined at present by a subjective evaluation or comparative measurements.

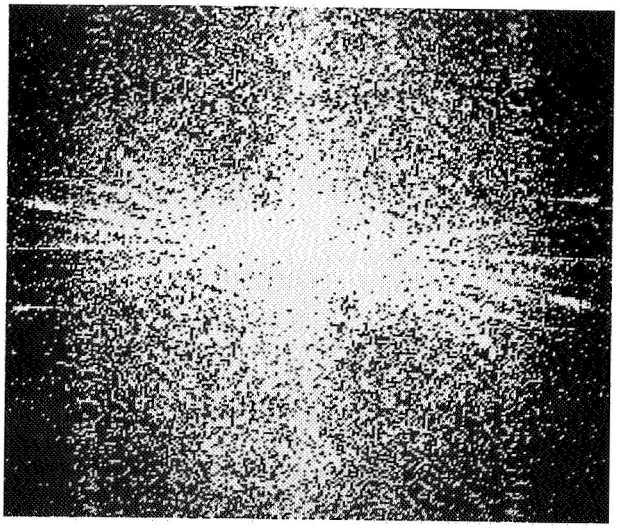
Transform threshold coding experiments have been performed for the Fourier and Hadamard transforms. Figures 5-7 and 5-8 show the location of samples above a threshold and the corresponding reconstructions



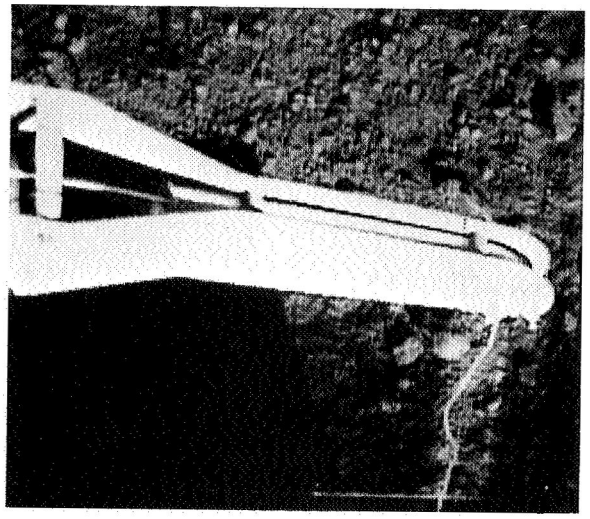
a) Map of Samples above 500



b) Fourier Transform of 500-level Thresholded Samples

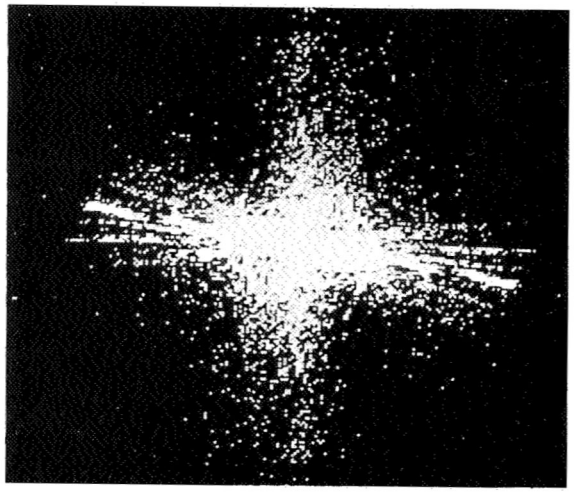


c) Map of Samples Above 100

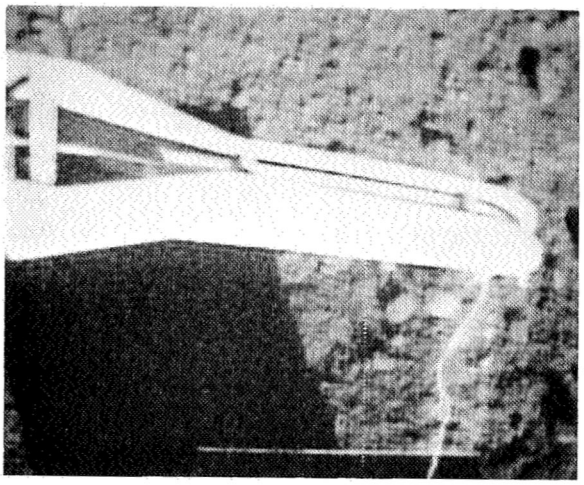


d) Fourier Transform of 100-level Thresholded Samples

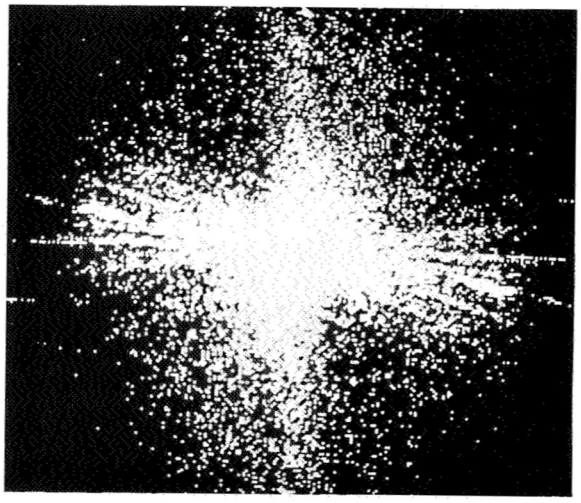
Figure 5-7. Fourier Transform Threshold Sampling with High and Low Thresholds - Boom



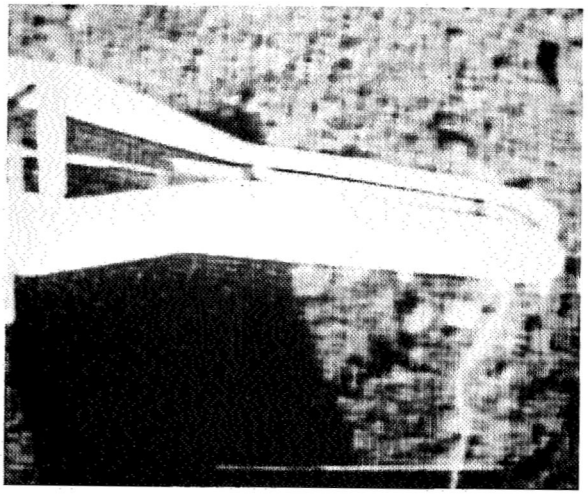
a) Map of Samples Above 300



b) Fourier Transform of 300-level Thresholded Samples



c) Map of Samples Above 200



d) Fourier Transform of 200-level Thresholded Samples

Figure 5-8. Fourier Transform Threshold Sampling with Intermediate Thresholds - Boom

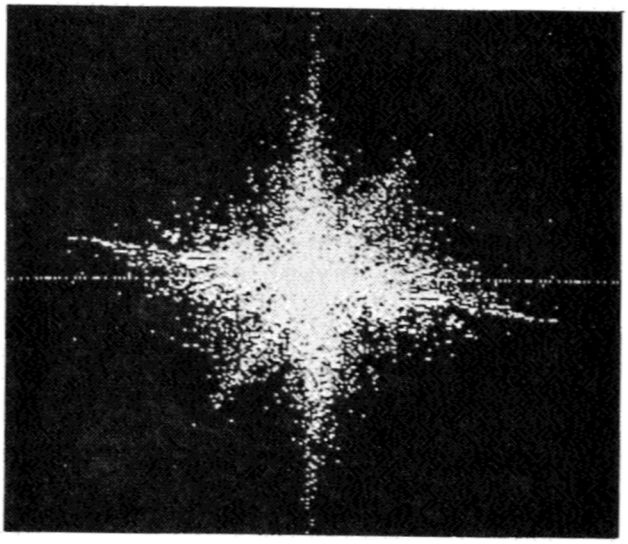
from these samples for the Fourier transform of the boom scene. If the threshold becomes too high, the loss of the high spatial frequency samples becomes noticeable. For this scene a threshold level of 200 provides good quality reconstructions. In this particular scene the magnitude of the largest, non-zero, spatial frequency is 53,186. The number of Fourier domain samples in the half plane above the threshold, and the equivalent sample reduction, are listed below.

Threshold Level	Number of Fourier Domain Samples Above Threshold	Sample Reduction
100	13,402	2.4
200	4,887	6.2
300	2,532	12.9

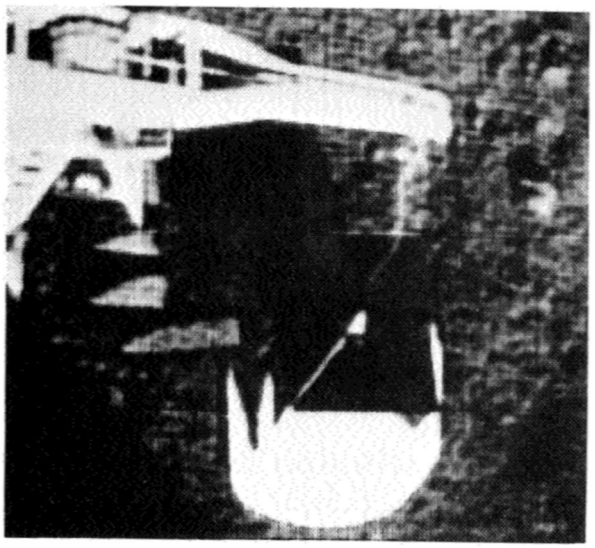
Figure 5-9 illustrates the results of the Fourier transform threshold coding experiment for the box scene. Again, a threshold level of 200 provides a good reconstruction.

Similar results have been obtained for threshold coding of Hadamard transform samples. Figure 5-10 shows maps of the location of significant samples of the Hadamard transform of the box scene.

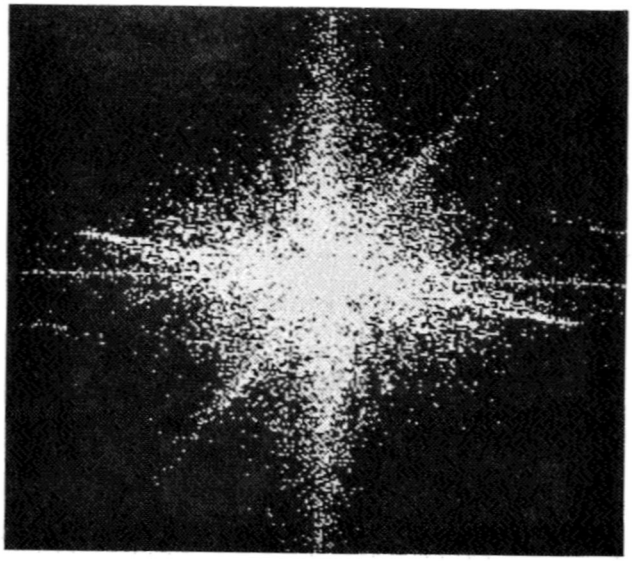
In order to achieve a bandwidth reduction with this threshold technique of sample deletion it is necessary to code the positions of the significant samples as well as their values. Position coding, of course, adds to the transmission bandwidth. Statistical data has been obtained on the number and location of significant samples in order to determine



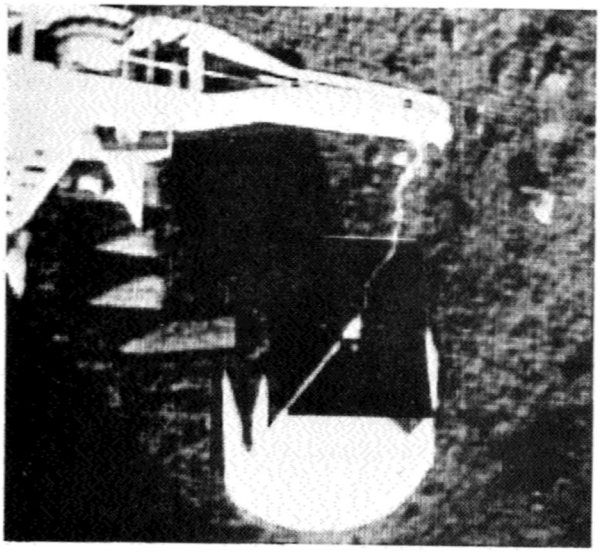
a) Map of Samples Above 300



b) Fourier Transform of 300-level Thresholded Samples

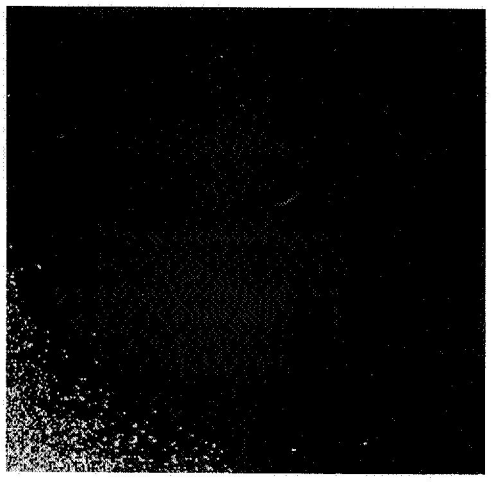


c) Map of Samples Above 200



d) Fourier Transform of 200-level Thresholded Samples

Figure 5-9. Fourier Transform Threshold Sampling with Intermediate Thresholds - Box



a) Map of Samples Above 800

b) Map of Samples above 400

Figure 5-10. Hadamard Transform Threshold Sampling with Intermediate Thresholds - Box

useful codes and evaluate the amount of bandwidth reduction possible. Figure 5-11 is a plot of run lengths of significant samples of the Fourier transform of the box for the threshold set at level 200. For this scene the number of run lengths greater than 16 elements is small so that the run lengths can be truncated to 16 without appreciably affecting the distribution. For a sample run length position code with a constant word length of 4 bits, a bandwidth reduction of greater than 4 : 1 is possible for this scene. A Huffman variable length code would result in a slightly higher bandwidth reduction factor.

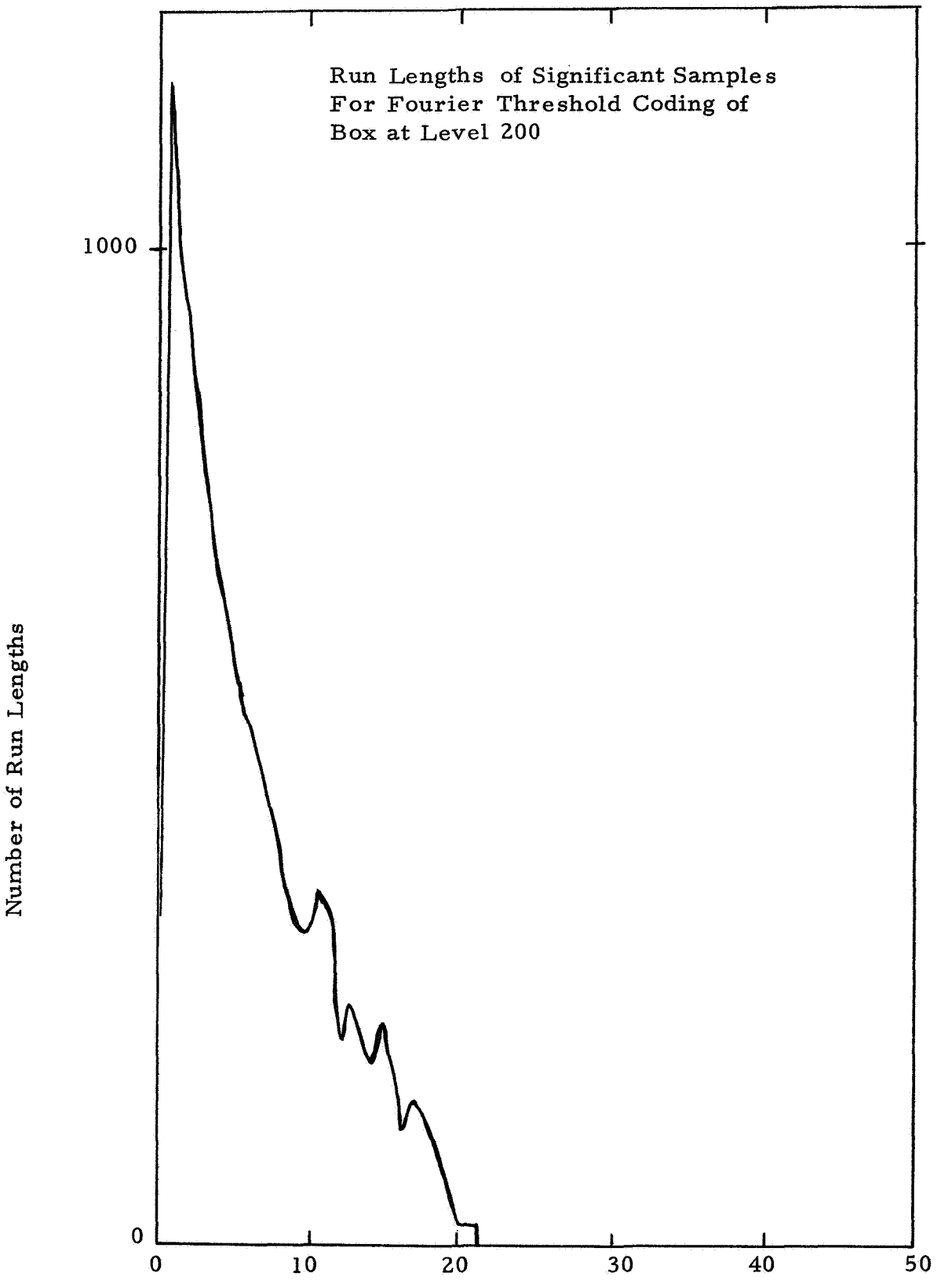


Figure 5-11. Run lengths of Significant Samples for Fourier Transform Threshold Sampling of Box at Level 200.

6. Error Tolerance

The major advantage of image transform coding other than its potential for bandwidth reduction is the tolerance to channel errors that transform coding affords. The inherent "error averaging" property of transform coding combined with error correction coding of transform samples provides a means of image coding for which channel errors are less deleterious than for conventional spatial coding of an image.

To illustrate the error tolerance feature of transform coding a binary symmetric channel will be assumed as a model for the channel. In the binary symmetric channel shown in figure 6-1 the probability of receiving an incorrect symbol is given by p for the transmission of ones or zeros.

6.1 Channel Noise Effects

An intuitive justification for transmitting the frequency rather than the spatial domain of an image is the fact that for many transforms the channel noise introduced in the transform of an image tends to be distributed evenly over the entire reconstructed image. Consequently the noise manifests itself as a low frequency effect in reconstruction. Since the eye is more sensitive to the high frequency "salt and pepper" effect of channel noise in the spatial domain, the same channel noise power in the frequency domain is somewhat less offensive. Figure 6-2a shows a mid-grey scene after having passed through a channel with probability of error of 0.1. Figure 6-2b is the Fourier transform of the output of the same channel whose input was the Fourier transform of

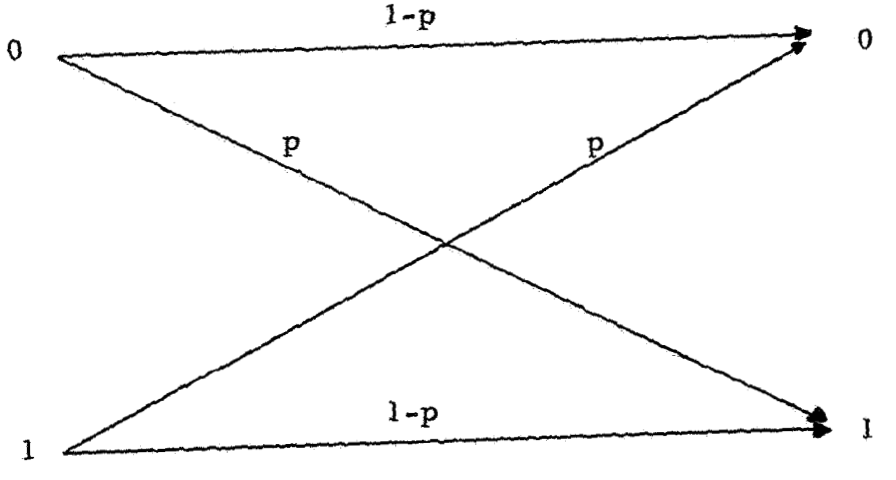
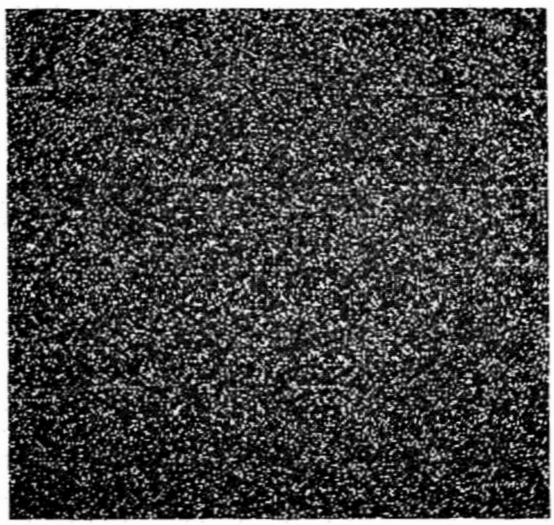
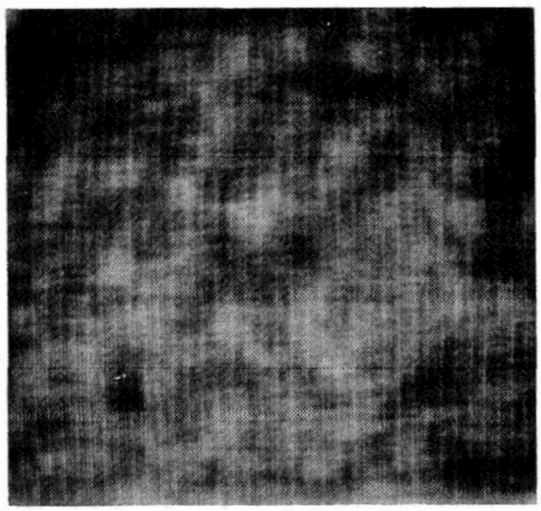


Figure 6-1. Model of a Binary Symmetric Channel



a) BSC Noise in Spatial Domain

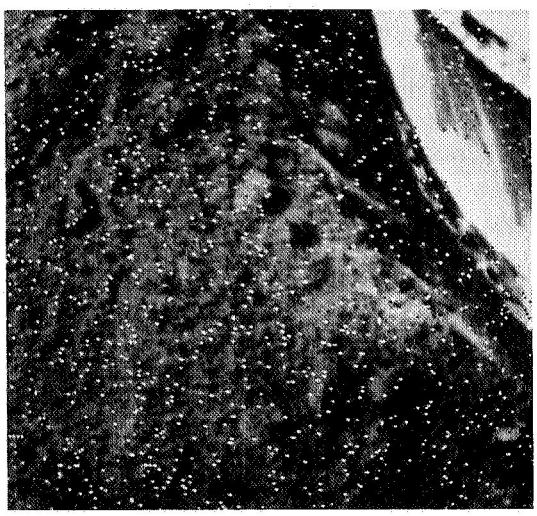


b) Fourier Transform of BSC in Fourier Domain

Figure 6-2. Binary Symmetric Channel Noise with Error Rate $p = 10^{-1}$

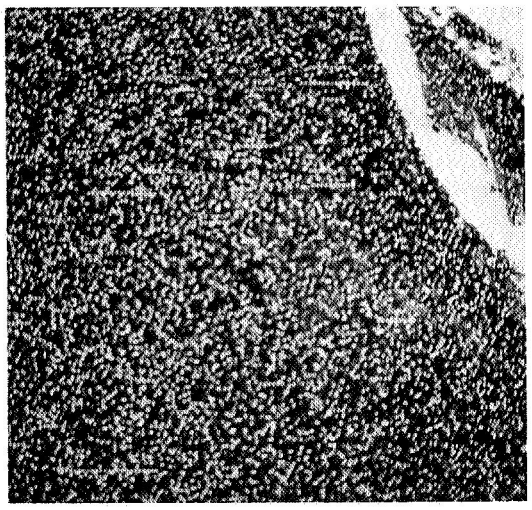
the mid-grey scene. Both scenes have the same amount of noise energy but that energy is distributed quite differently. A quantizing and coding method can be developed to take advantage of the inherent high frequency or "salt and pepper" noise immunity that Fourier domain coding offers. As a first step in this direction, a requirement will be made that each quantum level occur equally likely as any other quantum level. This quantization criterion will guarantee that each code word is equally likely to occur and will avoid any unexpected noise biasing, since the binary symmetric channel effects each code bit, and therefore each code word, independently of all others.

Figure 6-3 contains a series of experimental results for the Fourier transform of the footpad scene using the Gaussian quantization law with the transform domain variance changing as a function of frequency according to the power spectrum of the original scene. The footpad and its quantized Fourier transform are passed through the same binary symmetric channel for two different error probabilities. These pictures are presented to demonstrate a further complication that must be avoided. The frequency induced noise energy is concentrated in low frequency variations which are so large that the high frequency information is lost due to normalization in reconstruction. This can be explained by the fact that the absolute, as opposed to the relative value of a bit error is much larger in the regions where the power spectrum is large. In the power spectrum of most images, the larger values occur at the lower frequencies, and thus the lower frequency noise errors have a greater

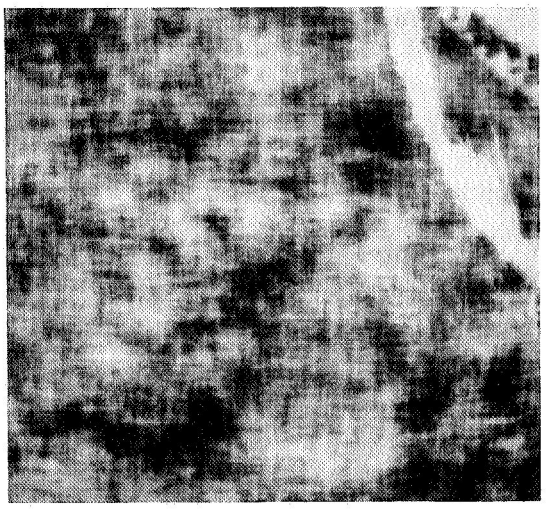


a) 10^{-3} Error Rate in the Spatial Domain

b) 10^{-3} Error Rate in the Fourier Domain



c) 10^{-1} Error Rate in the Spatial Domain



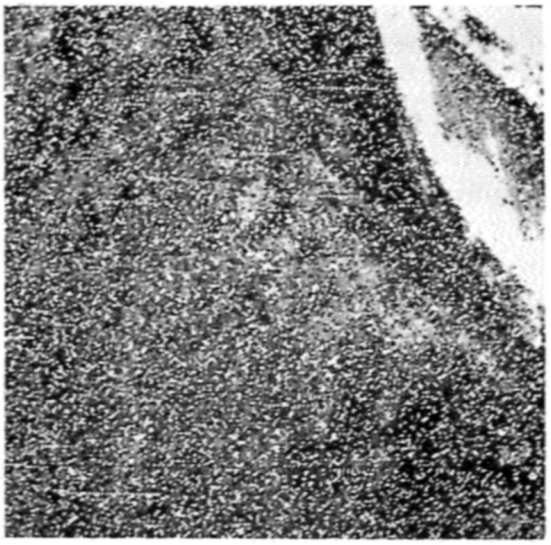
d) 10^{-1} Error Rate in the Fourier Domain

Figure 6-3. Binary Symmetric Channel Noise in Spatial and Fourier Domain Transmission

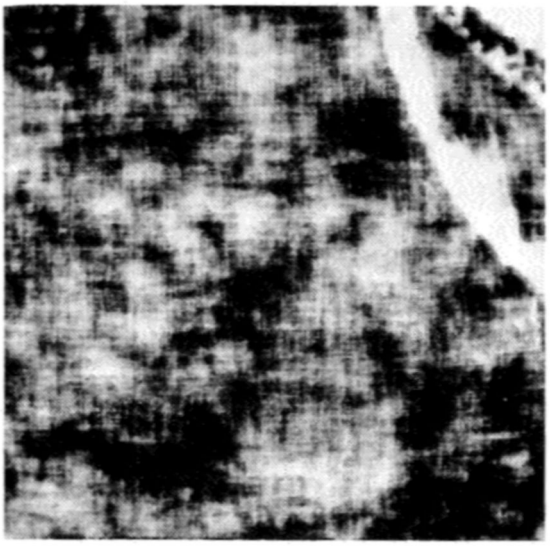
effect on the reconstructed image in the spatial domain. Further demonstration of this effect is afforded by figure 6-4. Figures 6-4a and 6-4b are the footpad noise scenes with error rates of 10^{-1} introduced in the space and frequency domain respectively. Figure 6-4c is the result of the same error rate channel noise in the frequency domain but with 20 x 40 or 800 of the lowest spatial frequencies transmitted error free. It is evident from figure 6-4c that the noise energy is now concentrated in the higher frequencies. Figure 6-4d has the lowest 6500 spatial frequencies transmitted error free.

6.2 Error Correction Transform Coding

As a result of the statistical regularity of samples in the frequency domain, a much smaller amount of error correction in this domain will yield a far better noise immunity than the same amount of error correction in the spatial domain. The nature of the quantization law is such that errors in certain positions of the frequency domain are much more bothersome than in other positions due to the large statistical variance of samples at these frequencies. Therefore, it is natural to develop an error correction rule to correct for errors only in these large variance regions. One such rule would be to error correct code those frequency samples which correspond to positions in the frequency domain where the power spectrum of the covariance function indicates a high probability of large sample values. This technique alone requires an increase in bandwidth to facilitate the error correction. However, it has been found



a) 10^{-1} Error Rate in the Spatial Domain



b) 10^{-1} Error Rate in the Fourier Domain

c) Reconstruction with the 800 Lowest Spatial Frequencies Errorless

d) Reconstruction with the 6500 Lowest Spatial Frequencies Errorless

Figure 6-4. Effect of Low Frequency Errors

that the small increase in bandwidth in the Fourier domain will result in far better reconstructions than the same increase in the spatial domain.

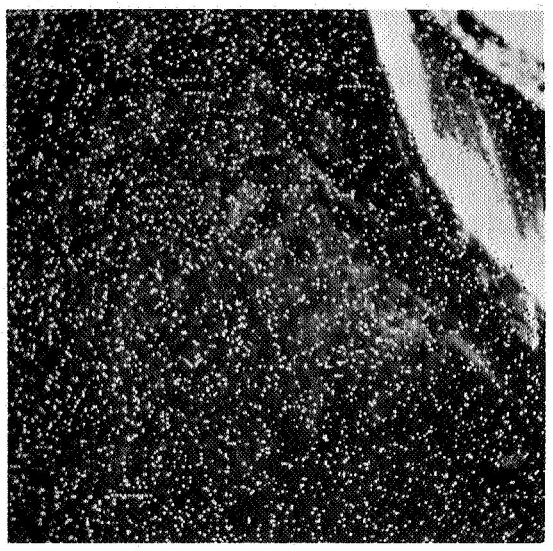
It is important to emphasize that the coding technique used for the Fourier domain should be tailored to a particular channel capacity. If the channel noise has an error rate less than about 10^{-3}, then it appears that no error correction is necessary as in figure 6-3b. However, under the circumstances of a high error rate, it often becomes more desirable to transmit as many error corrected samples as possible at the expense of not transmitting the entire frequency plane. Using such a system, corrected, but not necessarily errorless, data could be received until normal picture bandwidth has been reached, at which time transmission is terminated. In order to implement such a scheme, an error correcting code must be selected. The code selected will depend on how much of the frequency domain will be omitted due to the increased error correcting capability of the code. The main point of this discussion is to illustrate the variety of coding implementations possible for different channel conditions.

A specific example of the potential of the Fourier coding technique is presented below. A high error rate channel is assumed with rate $p = 4 \times 10^{-2}$. The equal bandwidth criterion is assumed. Consequently, the Fourier coding technique requires exactly the same bandwidth as conventional spatial domain transmission systems. The error correcting code must have at least six information bits. Two such codes which become candidates for implementation are a first order Reed Muller code and a

Bose Chaudhuri-Hocquenghem (BCH) code [42; 43, pg. 163]. The particular Reed Muller code of interest is a (32,6) code in which the minimum distance between code words is sixteen, and therefore, the code is capable of correcting a total of seven errors. The BCH code is a (31,6) code and is also capable of correcting seven errors. The BCH code will be used in the following discussion. Utilizing an error correcting code capable of seven error corrections does not mean that the six information bits will be received over the noisy channel error free. Since each code word length has been increased to thirty-one bits, eight or more errors per code word cannot be guaranteed to be corrected. The probability of having eight or more errors in the BCH code is given by the partial sum of the binomial distribution

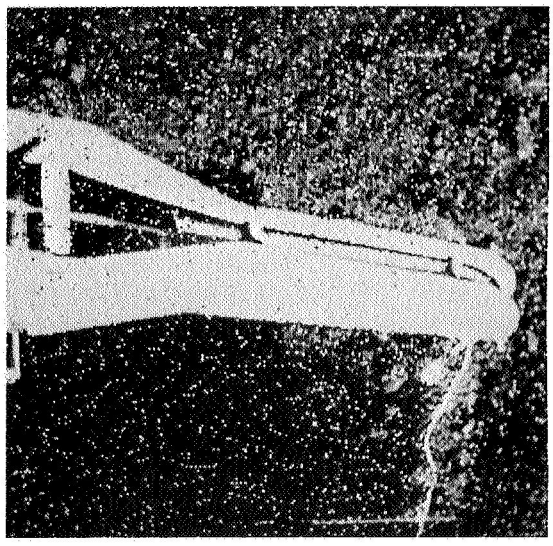
$$P(8 \text{ or more errors}) = \sum_{i=8}^{31} \binom{31}{i} p^i (1-p)^{31-i} \qquad (6\text{-}1)$$

where p is the binary symmetry channel error rate. This probability is an upper bound for the incorrect reception of a code word since the possibility of correct reception for greater than seven errors still exists but is unknown. For the specific channel error rate of 4×10^{-2}, the error corrected data samples will be received with probability of error no greater than 2.26×10^{-5} [44]. Figure 6-5 displays the results of this error correcting procedure. Figure 6-5a and 6-5c are two test scenes whose spatial domains are transmitted through the binary symmetric channel with the above error rate. Figure 6-5b and 6-5d

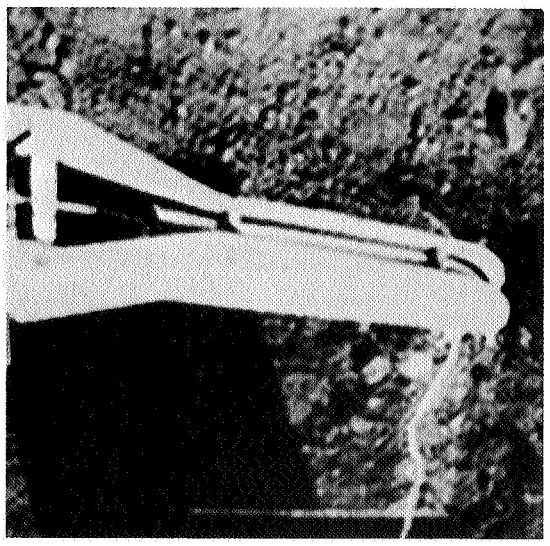


a) 4×10^{-2} Error Rate in the Spatial Domain

b) Error Corrected Retransformation



c) 4×10^{-2} Error Rate in the Spatial Domain



d) Error Corrected Retransformation

Figure 6-5. Equal Bandwidth Error Correction Technique

are the error correction Fourier domain transmission results for each of the test scenes. While there is a loss of high frequency information in figures 6-5b and 6-5d, there is a marked improvement over the spatial coding in figures 6-5a and 6-5c. It is evident that this particular type of coding offers a considerable advantage for very noisy communication channels.

7. Conclusions and Recommendations

The conclusions and recommendations of this study on the "Transform Processing and Coding of Images" are listed below.

(a) The Fourier and Hadamard image transforms are potentially useful for image coding. With threshold coding of transform samples, bandwidth reductions of 4 : 1 or greater are achievable for the Fourier and Hadamard transforms. Error correction of the low spatial frequency samples of the Fourier and Hadamard transforms provides a considerable improvement in the tolerance to channel errors for the same transmission bandwidth, as compared to conventional spatial domain coding.

(b) The Kronecker matrix transforms and the Karhunen-Loeve transform appear to possess many desirable properties for transform image coding, and therefore, deserve further investigation.

(c) An analysis of the statistical properties of the Fourier and Hadamard transforms indicates that the probability density of the transform samples tends toward a Gaussian distribution with a variance proportional to the power spectral density of the original image. Measurements over an ensemble of images should be made to verify this contention.

(d) Transform domain quantization rules have been developed for subjective viewing and mean square spatial error criteria

for the Fourier and Hadamard transforms. Samples of these transforms can be quantized with six bits per sample component without serious quantization error.

Appendix A

Hadamard Transform Algorithm

Computation of the one dimensional Hadamard transform by brute force methods requires N^2 operations where an operation is either an addition or subtraction. An algorithm for obtaining the one dimensional Hadamard transform in $N \log_2 N$ operations has been developed. Computational savings are realized by storage of intermediate results. A "fast" Hadamard transform algorithm was outlined in 1937 by Yates [30]. In 1958 Good described a matrix decomposition technique which can be implemented to perform a Hadamard transform with $N \log_2 N$ operations [29].

Figure A-1 illustrates the computations performed for our one dimensional Hadamard transformation with eight data points. The data points are arranged in a column at level 3 and then summed by pairs to produce intermediate results for level 2. A dotted line linking two nodes indicates that the data point at the higher level is multiplied by minus one before addition, or equivalently, the data point forms the subtrahend of a subtraction operation. Operations follow the tree graph to level 0 which is the ordered Hadamard transform of f(x). There are two operations performed at each node of levels 1, 2, and 3 yielding a total of $8\log_2 8 = 24$ operations.

The fast Hadamard transform algorithm performs all of the operations indicated in Figure A-1, but in a certain selected order. All operations of level k are not completed before proceeding to level k-1,

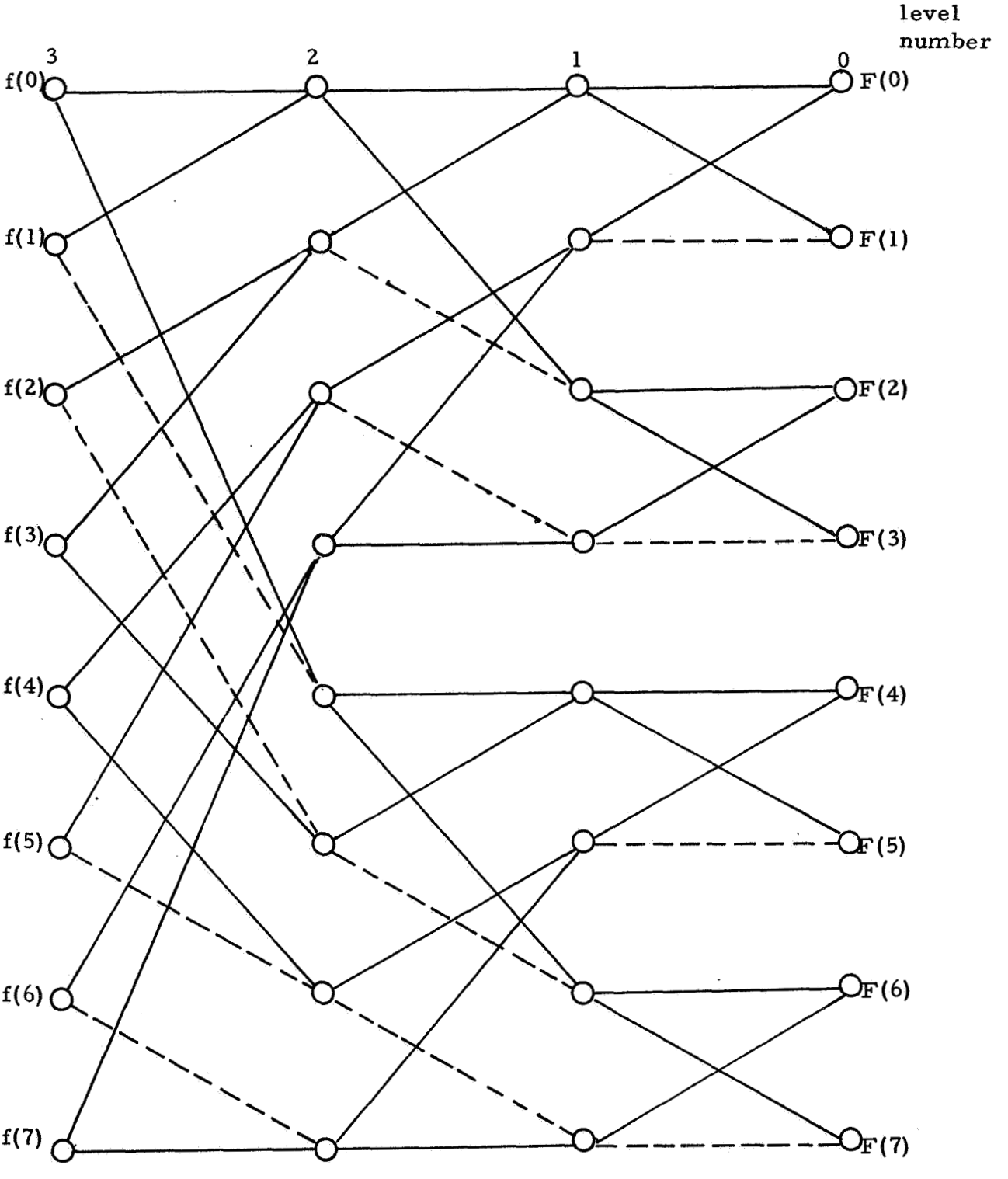


Figure A-1. Computations for one dimensional third order Hadamard transform

but rather operations are performed according to a sieving sequence. Figure A-2 describes the basic sequence computations. The first sequence is the "S" sequence in which the sum of all data points is formed to produce F(0). A "1" sequence subtracts the lower node from the upper node of a pair of nodes at level k-1 to produce a result at level k. In the "2" sequence, operations begin at level k-2 where pairs are subracted from one another to produce the results of level k-1 which in turn are added together. The "3" sequence and higher sequences to the "n" sequence follow directly. Figure A-2 also indicates the storage requirements for the computational procedure.

Original data is stored in a block of N words corresponding to level n. Intermediate results are stored n-1 different blocks of sizes 2^{n-1} to 2^2 words. These storage locations correspond to levels n-1 to 1 in the computation procedure.

Computation of the Hadamard transform begins with the "S" sequence which computes F(0). Subsequent calculations are controlled by the following sieving sequence of integers building up to integer n.

$$\{1,2,1,3,1,2,1,4,1,2,1,3,1,2,1,\cdots,1,n,1,2,1,\cdots,1,2,1,3,1,2,1\}$$

For example, after the "S" sequence, the "1" sequence computes F(1) using the intermediate results stored in level 1. Then, the "2" sequence computes F(2) using the intermediate results in level 2.

Figure A-3 gives the storage locations for computation of a one dimensional third order Hadamard transform. The computer operations

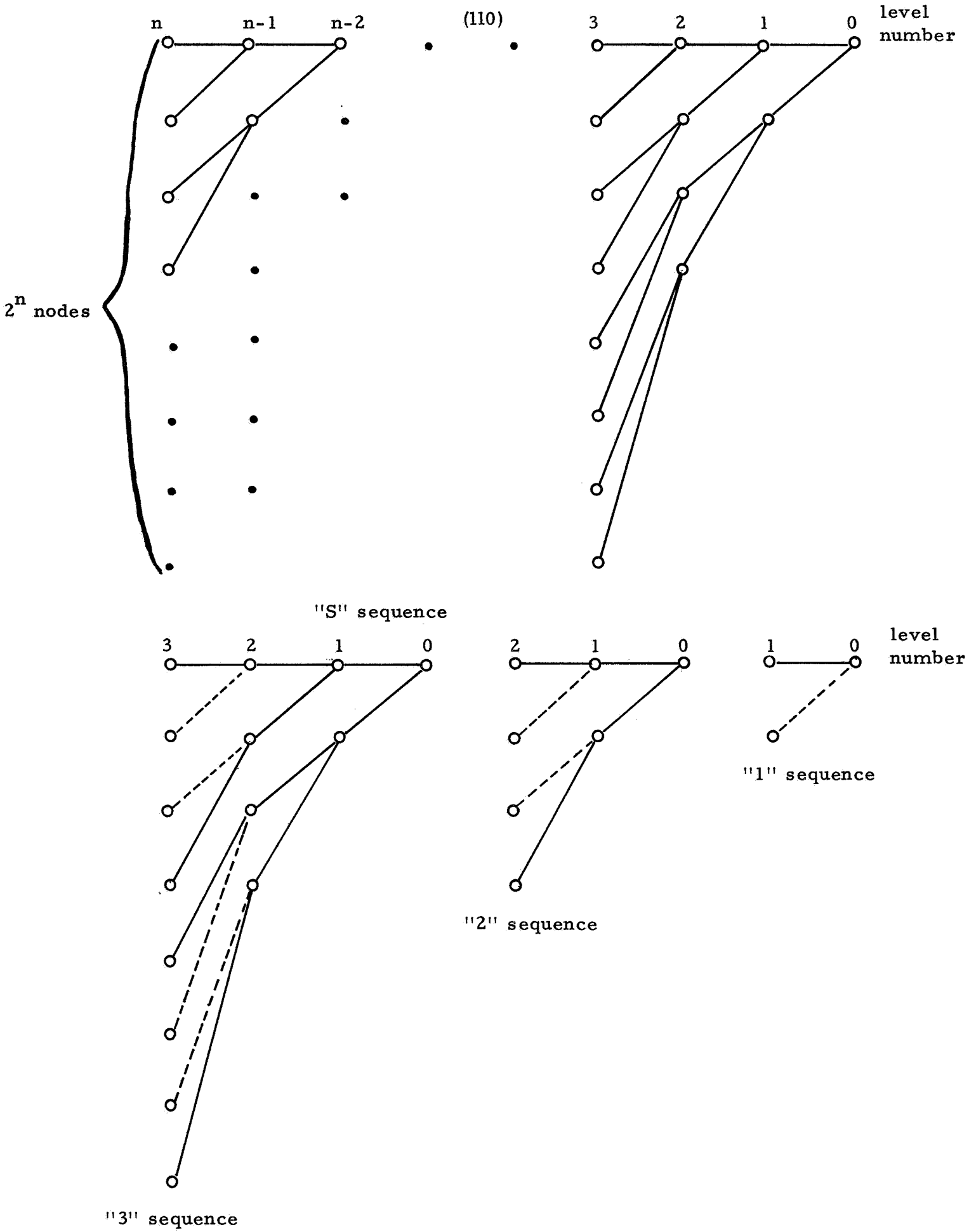


Figure A-2. Hadamard transform computational sequences.

	Level 3	2	1	0	
f(0)	A	I	M	F(0)	P
f(1)	B	J	N	F(1)	Q
f(2)	C	K		F(2)	R
f(3)	D	L		F(3)	S
f(4)	E			F(4)	T
f(5)	F			F(5)	U
f(6)	G			F(6)	V
f(7)	H			F(7)	W

Figure A-3. Storage Locations for Computation of One-Dimensional Third Order Hadamard Transform

for this example are listed in Figure A-4.

Since computation of the Hadamard transform requires only real additions and subtractions, whereas the Fourier transform computations are composed of complex multiplications, additions, and subtractions, considerable computational savings are afforded with the Hadamard transform. Both the Fourier and Hadamard transforms have been programmed on a TRW-530 digital computer. For a 256 by 256 point scene the Fourier transform can be computed in 20 minutes and the Hadamard transform in 3 minutes.

number	number	Operation	Contents of storage locations	Result
—	3	—	f(0) A f(1) B f(2) C f(3) D f(4) E f(5) F f(6) G f(7) H	
S	2	A + B → I	f(0) + f(1) I	
		C + D → J	f(2) + f(3) J	
		E + F → K	f(4) + f(5) K	
		G + H → L	f(6) + f(7) L	
	1	I + J → M	f(0) + f(1) + f(2) + f(3) M	
		K + L → N	f(4) + f(5) + f(6) + f(7) N	
	0	M + N → P	f(0) + f(1) + f(2) + f(3) + f(4) + f(5) + f(6) + f(7) P	F(0)
1	0	M - N → Q	f(0) + f(1) + f(2) + f(3) - f(4) - f(5) - f(6) - f(7) Q	F(1)
2	1	I - J → M	f(0) + f(1) - f(2) - f(3) M	
		L - K → N	-f(4) - f(5) + f(6) + f(7) N	
	0	M + N → R	f(0) + f(1) -f(2) - f(3) - f(4) - f(5) + f(6) + f(7) R	F(2)
1	0	M - N → S	f(0) + f(1) - f(2) - f(3) + f(4) + f(5) - f(6) - f(7) S	F(3)
3	2	A - B → I	f(0) - f(1) I	
		D - C → J	-f(2) + f(3) J	
		E - F → K	f(4) - f(5) K	
		H - G → L	-f(6) + f(7) L	
	1	I + J → M	f(0) - f(1) - f(2) + f(3) M	
		K + L → N	f(4) - f(5) - f(6) + f(7) N	
	0	M + N → T	f(0) - f(1) - f(2) + f(3) + f(4) - f(5) - f(6) + f(7) T	F(4)
1	0	M - N → U	f(0) - f(1) - f(2) + f(3) - f(4) + f(5) + f(6) - f(7) U	F(5)
2	1	I - J → M	f(0) - f(1) + f(2) - f(3) M	
		L - K → N	-f(4) + f(5) - f(6) + f(7) N	
	0	M + N → V	f(0) - f(1) + f(2) - f(3) - f(4) + f(5) - f(6) + f(7) V	F(6)
1	0	M - N → W	f(0) - f(1) + f(2) - f(3) + f(4) - f(5) + f(6) - f(7) W	F(7)

Figure A-4. Computer Operations for Computation of One-Dimensional Third Order Hadamard Transform

Appendix B

Fourier Transform Algorithm

The algorithm described here utilizes a modification to the Cooley-Tukey approach and requires N log N complex additions and only $\frac{N}{2}$ (log(N) - 2) + 1 complex multiplications for the one dimensional example. This savings is significant when one realizes that a complex multiplication includes two real multiplications and four real additions. Since most computers have a longer multiply time than add time, the computation time is greatly reduced. The reduction of complex multiplication operations can be achieved by evaluating spectral components in a specified order and using the fact that

$$\exp\left\{\frac{2\pi i}{N} k\right\} = -\exp\left\{\frac{2\pi i}{N}\left(k + \frac{N}{2}\right)\right\} \qquad \text{(B-1)}$$

The algorithm can be explained by letting f(x) be a one-dimensional complex function which has been sampled and stored in $N = 2^n$ locations and defining F(u) to be the spectral domain representation of the Fourier transform of f(x) given by the equation

$$F(u) = \sum_{x=0}^{N-1} f(x) \exp\left\{\frac{2\pi i}{N} xu\right\} \qquad \text{(B-2)}$$

Now expressing x and u in binary form,

$$x = x_{n-1}x_{n-2} \cdots x_1x_0; \quad x_j \in (0,1) \qquad \text{(B-3)}$$

$$u = u_{n-1}u_{n-2} \cdots u_1u_0; \quad u_j \in (0,1) \qquad \text{(B-4)}$$

and taking advantage of the integer periodic qualities of the complex exponential function, we can write

$$F(u_{n-1},\ldots,u_0) = \sum_{x_0=x_1\cdots x_{n-2}=x_{n-1}}^{1}\sum^{1}\cdots\sum^{1}\sum^{1} f(x_{n-1},\ldots,x_0)P_{n-1}P_{n-2}\cdots P_1P_0 \tag{B-5}$$

where

$$P_j = \exp\left(\frac{2\pi i}{N}\left[u_{n-1-j},\ldots,u_0,0,\ldots,0\right]x_j\right) \tag{B-6}$$

Now upon summing out the x_j, starting with the most significant bit (MSB), x_{n-1}, it is evident that the sum of equation (B-2) is made in a specific order. It is the computation in this order that allows for the storage of calculations so that no identical computation need be repeated.

However, now note that for each sum over x_j the exponential, P_j, can only take on specific values as a function of the particular spectral point, u, being evaluated. In fact, as more and more x_j are summed (approaching the LSB of the binary representation of x) it is evident that the exponential, P_j, takes on values defined by the shifting of the binary representation of the spectral point, u. For example, if the spectral point 001 were being evaluated, the P_j would take on values defined by

$$P_j = \exp\left[\frac{2\pi i}{N}(100)x_j\right], \quad P_j = \exp\left[\frac{2\pi i}{N}(010)x_j\right]$$

and

$$P_j = \exp\left[\frac{2\pi i}{N}(001)x_j\right].$$

For convenience, define $C_k \triangleq \exp\left\{\frac{2\pi ik}{N}\right\}$ and $C_0 = 1$. Now a table can be formed in which the P_j are defined for each step in the computation and for each spectral point. In constructing such a table, it is advantageous to list the spectral points as increasing binary numbers with the MSB on the right. In evaluating a particular spectral point, that point should be interpreted as its binary representation with the MSB on the left. For example, when n = 4, $N = 2^n = 16$ the table is given in Figure B-1. Note that such a table need not be constructed in the actual program.

In evaluating spectral components as they are listed in the table, it is found that after evaluating point u = 0 and retaining all intermediate sums in their storage locations the evaluation of point u = 8 requires only a single subtraction. This is because $C_j = C_{\left(j+\frac{N}{2}\right)}$ and from the table $C_8 = -C_0 = -1$. To evaluate spectral point u = 4 it is only necessary to back up two storage locations and perform a subtraction to obtain C_8 in column P_1 and then perform one multiplication to obtain C_4 in column P_0. Note that this C_4 is circled. To evaluate spectral point u = 12 a simple subtraction is necessary as $C_4 = -C_{12}$. Continuing in this manner it is seen that all columns contain pairs of $C_j = -C_{\left(j+\frac{N}{2}\right)}$ and that whenever one of these pairs is encountered, a simple subtraction is in order. The circled constants are the locations in the algorithm in which a multiplication must take place. The number of actual multiplications that will be necessary for each circled constant equals two raised to the index of the column in which the circled constant appears.

Thus, in the table the total number of multiplications required is

Spectral Point		Exponential Factor Values				Program Number
	$u_3u_2u_1u_0$	P_3	P_2	P_1	P_0	PN
0	0 0 0 0	1	1	1	1	0
8	1 0 0 0	1	1	1	C_8	1
4	0 1 0 0	1	1	C_8	C_4	2
12	1 1 0 0	1	1	C_8	C_{12}	1
2	0 0 1 0	1	C_8	C_4	C_2	3
10	1 0 1 0	1	C_8	C_4	C_{10}	1
6	0 1 1 0	1	C_8	C_{12}	C_6	2
14	1 1 1 0	1	C_8	C_{12}	C_{14}	1
1	0 0 0 1	C_8	C_4	C_2	C_1	4
9	1 0 0 1	C_8	C_4	C_2	C_9	1
5	0 1 0 1	C_8	C_4	C_{10}	C_5	2
13	1 1 0 1	C_8	C_4	C_{10}	C_{13}	1
3	0 0 1 1	C_8	C_{12}	C_6	C_3	3
11	1 0 1 1	C_8	C_{12}	C_6	C_{11}	1
7	0 1 1 1	C_8	C_{12}	C_{14}	C_7	2
15	1 1 1 1	C_8	C_{12}	C_{14}	C_{15}	1

Figure B-1. Spectral Point Evaluation for N = 16

equal to $(7)2^0 + (3)2^1 + (1)2^2 = 17$ complex multiplications. In general, the number of complex multiplications is equal to:

$$\sum_{k=2}^{n} (2^{k-1} - 1)2^{n-k} = \frac{N}{2}(n-2) + 1$$

$$\text{where } N = 2^n \qquad \text{(B-7)}$$

As an example, Fourier transforming a function with 1024 points requires 4097 complex multiplications as compared to $N \log_2 N = 10,240$ complex multiplications. Note that with such an algorithm the number of exponential constant, C_j, that must be used is $\frac{N}{2} - 1$. In general, the number of storage locations for data for the entire transformation is

$$\begin{aligned} & N && \text{initial data points} \\ & N && \text{spectral data points} \\ & N \sum_{i=1}^{n-1} \left(\frac{1}{2}\right) && \text{temporary storage locations} \\ & = N-2 \end{aligned}$$

Total data locations = $N + N + N-2 = 3N - 2$.

Implementation of the algorithm is best described with reference to Figure B-2. In designing the computer program to implement the one-dimensional algorithm additional computation may be saved if the input is known to be real. Such a restriction on the input manifests itself as a symmetric conjugate property of the Fourier transform and consequently only $\frac{N}{2} + 1$ spectral data points need be calculated. However, the program must be able to accept complex inputs whenever the case

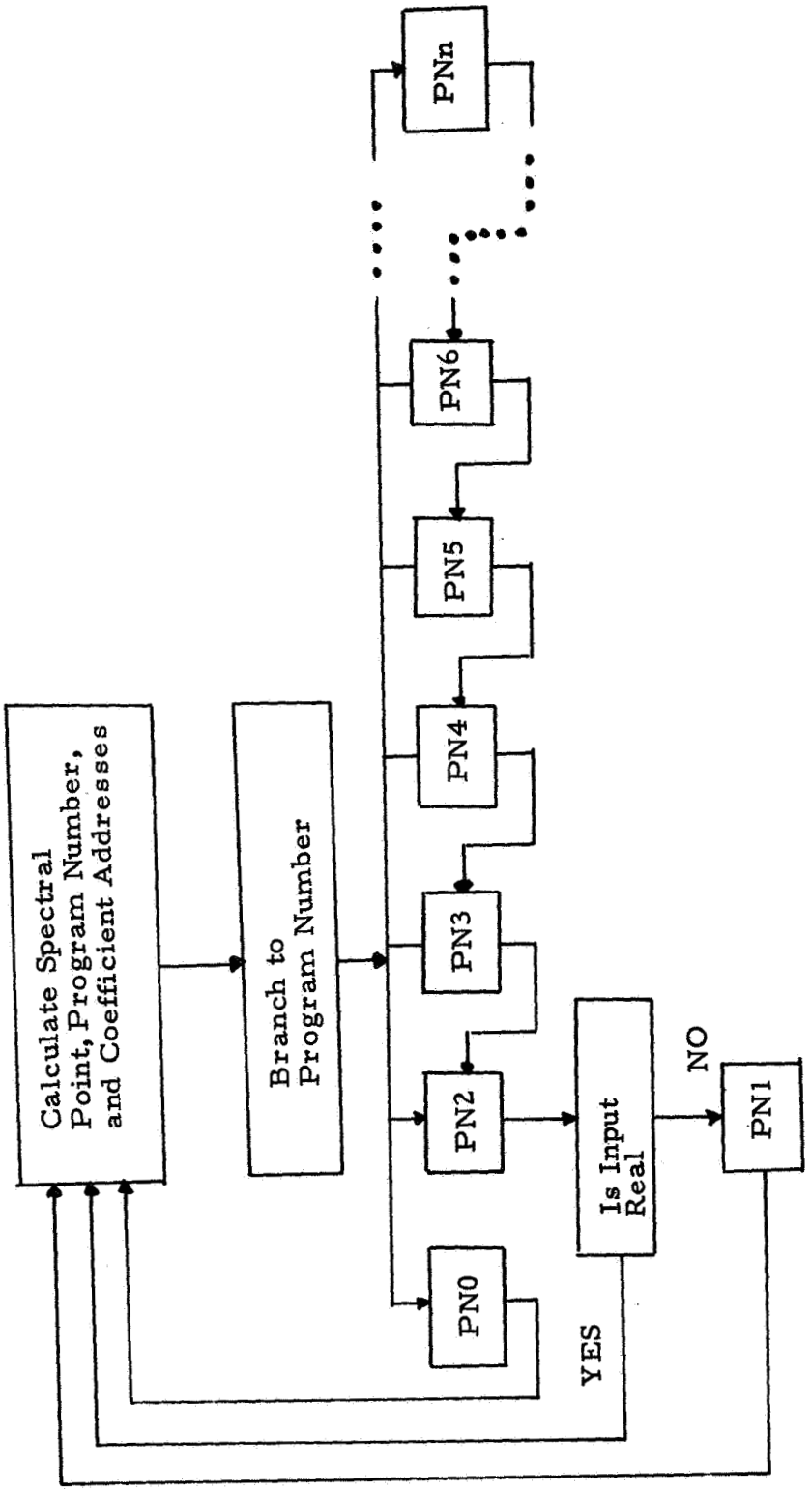


Figure B-2. Program Block Diagram

should arise. Figure B-2 shows the command control for the program. The spectral point is calculated by incrementing a binary counter and interpreting the most significant bit as the least. The program numbers are obtained by an inverse sieving operation and are exemplified in Figure B-1. The coefficient addresses are computed when a multiplication becomes necessary, also Figure B-1. Finally a branch to a particular program takes place. Program PN0 will always be the first program and simply sequentially adds the first half of the data input to the second half storing the results in $\frac{N}{2}$ storage locations. This operation is repeated until the final sum is stored in one storage location. This result is the amount of dc in the original function, f(x). Program PN1 simply backs up a storage location and performs a subtraction resulting in the value for the spectral point found by adding $\frac{N}{2}$ to the prior spectral point. This can be verified from Figure B-1 where program PN1 always evaluates the spectral point $\frac{N}{2} = 8$ greater than the previous spectral point. Notice that program PN1 always evaluates spectral points greater than or equal to $\frac{N}{2}$ and consequently need not be implemented, except once, for real inputs. All other programs back up a respective number of levels of storage locations, implement a series of subtractions and multiplications and branch to the next lower program number. An eight point transform program has been included to demonstrate the operations involved. The table for n = 3 is shown in Figure B-3. In Figure B-4 the eight spectral points are formed showing storage locations and redundant use of storage locations. Notice the fact that $C_k C_j = C_{k+j}$ has been used in some of the

Spectral Point		Exponential Factor Values			Program Number
	$u_2u_1u_0$	P_2	P_1	P_0	PN
0	0 0 0	1	1	1	0
4	1 0 0	1	1	C_4	1
2	0 1 0	1	C_4	C_2	2
6	1 1 0	1	C_4	C_6	1
1	0 0 1	C_4	C_2'	C_1	3
5	1 0 1	C_4	C_2	C_5	1
3	0 1 1	C_4	C_6	C_3'	2
7	1 1 1	C_4	C_6	C_7	1

Figure B-3. Spectral Point Evaluation for N = 8

PN	Spectral Pt.	Operations
PN0	u = 000	1+5 → 9
		2+6 → 10
		3+7 → 11
		4+8 → 12
		9+11 → 13
		10+12 → 4
	F(0)←	13+14
PN1	u = 100	
	F(4)←	13-14
PN2	u = 010	9-11 → 13
		10-12 → 14
		C_2x14 → 14
	F(2)←	13+14
PN1	F(6)←	13-14
PN3	u = 001	1-5 → 9
		2-6 → 10
		3-7 → 11
		4-8 → 12
		C_2x11 → 11
		C_2x12 → 12
		9+11 → 13
		10+12 → 12
		C_1x14 → 14
	F(1)←	13+14
PN1	u = 101	
	F(5)←	13-14
PN2	u = 001	9-11 → 13
		10-12 → 14
		C_3x14 → 14
	F(3)←	13+14
PN1	u = 111	
	F(7)←	13-14

$\boxed{f(0)}_1$ $\boxed{f(1)}_2$ $\boxed{f(2)}_3$ $\boxed{f(3)}_4$ $\boxed{f(4)}_5$ $\boxed{f(5)}_6$ $\boxed{f(6)}_7$ $\boxed{f(7)}_8$

$\boxed{f(0)+f(4)}_9$ $\boxed{f(1)+(5)}_{10}$ $\boxed{f(2)+f(6)}_{11}$ $\boxed{f(3)+f(7)}_{12}$

$\boxed{f(0)+f(4)+f(2)+f(6)}_{13}$ $\boxed{f(1)+f(5)+f(3)+f(7)}_{14}$

$\boxed{f(0)+f(4)+f(2)+f(6)+f(1)+f(5)+f(3)+f(7)}$

$\boxed{f(0)+f)4)+f(2)+f(6)+C_4f(1)+C_4f(5)+C_4f(3)+C_4f(7)}$

$\boxed{f(0)+f(4)+C_4f(2)+C_4f(6)}_{13}$ $\boxed{f(1)+f(5)+C_4f(3)+C_4f(7)}_{14}$

$\boxed{C_2f(1)+C_2f(5)+C_6f(3)+C_6f(7)}_{14}$

$\boxed{f(0)+f(4)+C_4f(2)+C_4f(6)+C_2f(1)+C_2f(5)+C_6f(3)+C_6f(7)}$

$\boxed{f(0)+f(4)+C_4f(2)+C_4f(6)+C_6f(1)+C_6f(5)+C_2f(3)+C_2f(7)}$

$\boxed{f(0)+C_4f(4)}_9$ $\boxed{f(1)+C_4f(5)}_{10}$ $\boxed{f(2)+C_4f(6)}_{11}$ $\boxed{f(3)+C_4f(7)}_{12}$

$\boxed{C_2f(2)+C_6f(6)}_{11}$ $\boxed{C_2f(3)+C_6f(7)}_{12}$

$\boxed{f(0)+C_4f(4)+C_2f(2)+C_6f(6)}_{13}$ $\boxed{f(1)+C_4f(5)+C_2f(3)+C_6f(7)}_{14}$

$\boxed{C_1f(1)+C_5f(5)+C_3f(3)+C_7f(7)}_{14}$

$\boxed{f(0)+C_4f(4)+C_2f(2)+C_6f(6)+C_1f(1)+C_3f(3)+C_7f(7)+C_5f(5)}$

$\boxed{f(0)+C_4f(4)+C_2f(2)+C_6f(6)+C_5f(1)+C_1f(5)+C_7f(3)+C_3f(7)}$

$\boxed{f(0)+C_4f(4)+C_6f(2)+C_2f(6)}_{13}$ $\boxed{f(1)+C_4f(5)+C_6f(3)+C_2f(7)}_{14}$

$\boxed{C_3f(1)+C_7f(5)+C_1f(3)+C_5f(7)}_{14}$

$\boxed{f(0)+C_4f(4)+C_6f(2)+C_2f(6)+C_3f(1)+C_7f(5)+C_1f(3)+C_5f(7)}$

$\boxed{f(0)+C_4f(4)+C_6f(2)+C_2f(6)+C_7f(1)+C_3f(5)+C_5f(3)+C_1f(7)}$

Total Adds: 24

Total Mult.: 5

FIGURE B-4

COMPUTER OPERATIONS AND STORAGE LOCATIONS
FOR N = 8

calculations. Also, included in Figure B-4 is a list of operations, the program controlling the operations, and destinations on the data in all locations.

Appendix C

Experimental Image Processor

Figure C-1 illustrates the experimental image processing equipment used for the images which appear in this report. For low resolution images the vidicon camera can be used to input video information into the A D converter for digitization. However, for higher resolution a flying spot scanner is required. The video signal is converted to digital form by an analog-to-digital converter, and stored on magnetic tape for processing.

A TRW-530 digital computer with 8,000 words of storage and 2 magnetic tape units forms the central processor. With this computer and the existing Fourier transform program of Appendix B a Fourier transform of a 256 by 256 sample point image with complex sample points can be performed in less than 20 minutes. With real sample points only, the computation time is reduced to about 15 minutes. Shorter computation time could be achieved, with a faster computer having greater storage capacity. The Hadamard program of Appendix A for the same resolution image can be implemented in 3 minutes. The transformed results are then modified according to some "filtering" function and the resulting plane is inverse transformed back to image form. The result is then stored on magnetic tape from which it is converted through a D/A device for display on a flying spot scanner or cathode ray tube monitor. The images in this report have all been photographed from the cathode ray tube monitor.

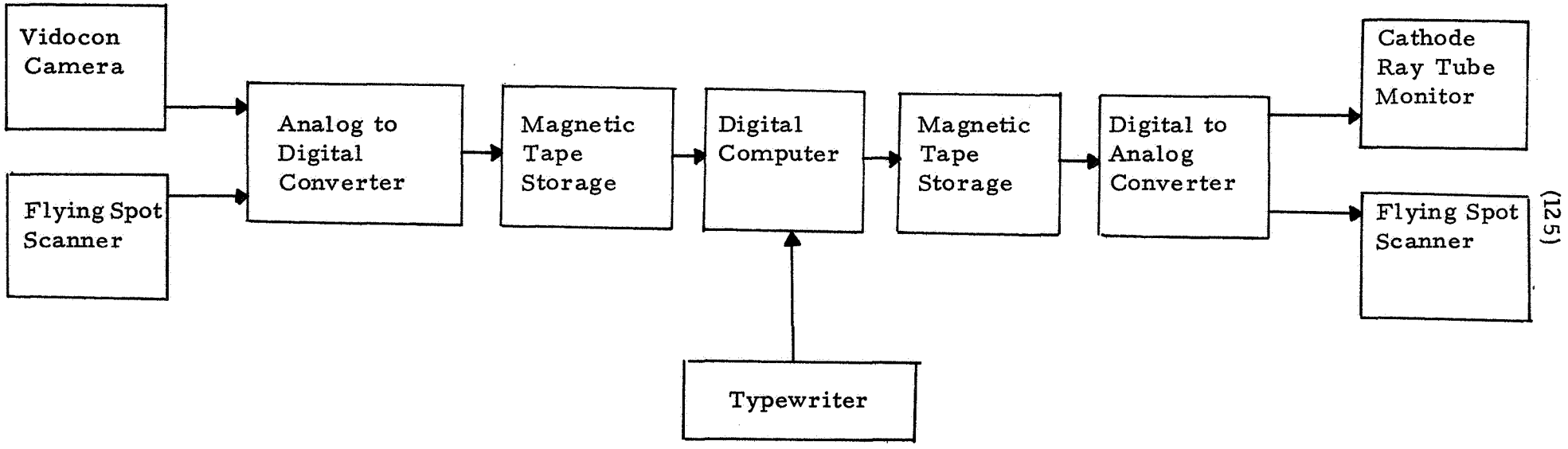


Figure C-1. Computer Image Processing Equipment

Appendix D

References

1. W. K. Pratt "A Bibliography on Television Bandwidth Reduction Studies," IEEE Transactions on Information Theory, Vol. IT-13, No. 1 (January, 1967), pp. 114-115.

2. A. Rosenfeld " Bandwidth Reduction Bibliography", IEEE Transactions on Information Theory, Vol. IT-14, No. 4 (July, 1968). PP. 601-602

3. Special issue on redundancy reduction Proceedings IEEE, Vol.55 , No.3 (March, 1967).

4. J. W. Cooley and J. W. Tukey "An Algorithm for the Machine Calculation of Complex Fourier Series," Mathematics of Computation, vol. 19, no. 90, (1956), pp. 297-301.

5. W. T. Cochran, et. al. "What is the Fast Fourier Transform?," Proceedings IEEE, vol. 55, no. 10, (October, 1967), pp. 1664-1673.

6. J. W. Cooley, P.A.W. Lewis, and P. D. Welch, "Historical Notes on the Fast Fourier Transform," Proceedings IEEE, vol. 55, pp. 1675-1677, October, 1967.

7. E. O. Brigham and R. E. Morrow "The Fast Fourier Transform," IEEE Spectrum, vol. 4, no. 12, (December, 1967), pp. 63-70.

8. H. C. Andrews, "A High Speed Algorithm for the Computer Generation of Fourier Transforms," IEEE Transactions on Computers, vol. C-17, no. 4, (April, 1968), pp. 373.

9. H. C. Andrews "Fourier Coding of Images," University of Southern California, Electronic Sciences Laboratory, USCEE Report No. 271 (June, 1968).

10. H. C. Andrews and W. K. Pratt "Fourier Transform Coding of Images," Hawaii International Conference on System Sciences, (January, 1968), pp. 677-679.

11. H. C. Andrews and W. K. Pratt "Television Bandwidth Reduction by Fourier Image Coding," Society of Motion Picture and Television Engineers, 103rd Technical Conference, (May, 1968).

12. H. C. Andrews and W. K. Pratt "Television Bandwidth Reduction by Encoding Spatial Frequencies," Journal Society of Motion Picture and Television Engineers, Vol. 77 (December, 1968), pp. 1279-1281.

13. W. K. Pratt, J. Kane, and H. C. Andrews "Hadamard Transform Image Coding," Proceedings IEEE, Vol. 57, No. 1 (January, 1969), pp.

14. H. C. Andrews and W. K. Pratt "Transformation Coding for Noise Immunity and Bandwidth Reduction," Second Annual Hawaii International Conference on System Sciences, (January, 1969).

15. W. K. Pratt and H. C. Andrews "Two Dimensional Transform Coding of Images," 1969 IEEE International Symposium on Information Theory, (January, 1969).

16. J. Hadamard "Resolution d'une Question Relative aux Determinants," Bulletin des Sciences Mathematiques," (2), vol. 17, Part 1, (1893), pp. 240-246.

17. H. J. Ryser, Combinatorial Mathematics, John Wiley, New York, (1963).

18. S. W. Golomb, et. al. "Digital Communications," Prentice-Hall, (1964).

19. R.E.A.C. Paley "On Orthogonal Matrices," Journal of Mathematics and Physics, Massachusetts Institute of Technology, vol. 12, (1933), pp. 311-320.

20. J. Williamson "Hadamard's Determinant Theorem and the Sum of Four Squares," Duke Mathematical Journal, vol. 11, (1944), pp. 65-81.

21. L. Baumert, S. W. Golomb, and M. Hall, Jr. "Discovery of an Hadamard Matrix of Order 92," Bulletin American Mathematical Society, vol. 68, (1962), pp. 237-238.

22. H. F. Harmuth "A Generalized Concept of Frequency and Some Applications," IEEE Transactions on Information Theory, vol. IT-14, no. 3, (May, 1968), pp. 375-382.

23. J. L. Walsh "A Closed Set of Orthogonal Functions," American Journal Mathematics, vol. 55, (1923), pp. 5-24.

24. N. J. Fine "On the Walsh Functions," Transactions American Mathematical Society, vol. 65, (1949), pp. 372-414.

25. N. J. Fine "The Generalized Walsh Functions," Transactions American Mathematical Society, vol. 69, (1950), pp. 66-77.

26. G. W. Morgenthaler "On Walsh-Fourier Series," Transactions American Mathematical Society, vol. 84, (1957), pp. 472-507.

27. K. W. Henderson "Some Notes on the Walsh Functions," IEEE Transactions on Electronic Computers, vol. EC-13, (February, 1964), pp. 50-52.

28. H. Rademacher "Einige Satze von Allgemeinen Orthogonal-Funktionen," Mathematics Annals, vol. 87, (1922), pp. 122-138.

29. I. J. Good "The Interaction Algorithm and Practical Fourier Analysis," Journal of the Royal Statistical Society, Series B, vol. 20, pp. 361-372, 1958.

30. F. Yates, The Design and Analysis of Factorial Experiments, Harpenden: Imperial Bureau of Soil Analysis, 1937.

31. H. P. Kramer and M. V. Mathews "A Linear Coding for Transmitting a Set of Correlated Signals," IRE Transactions on Information Theory, Vol. IT-2 (September, 1956), pp. 41-46.

32. J. E. Whelchel, Jr., and D. F. Guinn "The Fast Fourier-Hadamard Transform and Its Use in Signal Representation and Classification," EASCON 1968 Convention Record, (1968), pp. 561-573.

33. W. M. Gentleman, "Matrix Multiplication and Fast Fourier Transformations," Bell System Technical Journal, vol. 47, pp. 1099-1103, (July-August, 1968).

34. Loeve, M. "Probability Theory," D. Van Nostrand, Priceton, N.J. (1955).

35. Bochner, S. "Harmonic Analysis and the Theory of Probability," University of California Press, Berkeley, California (1955).

36. Gnedenko, B. V. "The Theory of Probability," Chelsea Publishing Company, New York, N.Y. (1962).

37. Rozanov, Y. A. "Stationary Random Processes," Holden-Day Publishing Company, San Francisco, California (1967).

38. Diananda, P. H. "The Central Limit Theorem for m-Dependent Variables," Proceedings of the Cambridge Philosophical Society, Vol. 51 (1955), pp. 92-95.

39. A Papoulis The Fourier Integral and Its Applications, McGraw-Hill, New York (1962).

40. H. C. Andrews "Entropy Considerations in the Frequency Domain," Proceedings IEEE Letters, vol. 46, no. 1, (January, 1968), pp. 113-114.

41. Panter, P. F. "Modulation, Noise, and Spectral Analysis Applied to Information Transmission," McGraw-Hill, New York, N.Y. (1965).

42. Reed, I. S. "A Class of Multiple Error-Correcting Codes and the Decoding Scheme," IRE Transactions, PGIT-4 (1954), pp. 38-49.

43. Peterson, W. W. "Error Correcting Codes," The M.I.T. Press, Cambridge, Massachusetts (1961).

44. "Tables of the Binomial Probability Distribution," Department of Commerce, National Bureau of Standards, Applied Mathematics Series No. 6 (January, 1950).

45. Davenport, W. B., Jr., and Root, W. L. "An Introduction to the Theory of Random Signals and Noise," McGraw-Hill, New York, N.Y. (1958).

46. W. R. Crowther and C. M. Rader, "Efficient Coding of Vocoder Channel Signals Using Linear Transformation," Proc. IEEE Letters, vol. 54, pp. 1594-1595, November, 1966.

47. P. F. Panter and W. Dite "Quantization Distortion in Pulse Count Modulation with Nonuniform Spacing of Levels," Proceedings IRE, Vol. 39, No. 1 (January, 1951), pp. 44-48.

48. J.J.Y. Huang and P. M. Schutheiss "Block Quantization of Correlated Gaussian Random Variables," IEEE Transactions on Communication Systems, Vol. CS-11, No. 3 (September, 1963), pp. 289-296.

49. T. Y. Young and W. H. Huggins "On the Representation of Electrocardiographs," IEEE Transactions on Bio-Medical Electronics, Vol. BME-10, No. 3 (July, 1963), pp. 86-95.

50. L. M. Goodman "A Binary Linear Transformation for Redundancy Reduction," Proceedings IEEE Letters, Vol. 55, No. 3 (March, 1967), pp. 467-468.

51. C. A. Andrews, J. M. Davies, and G. R. Schwarz "Adaptive Data Compression," Proceedings IEEE, Vol. 55, No. 3 (March, 1967), pp. 267-277.

52. C. J. Palermo, R. V. Palermo, and H. Horowitz "The Use of Data Omission for Redundancy Removal," Record International Space Electronics and Telemetry Symposium, (1965), pp. (11)D1-(11)D16.